Detectie van epileptische aanvallen: de Reservoir Computing-benadering

Detection of Epileptic Seizures: the Reservoir Computing Approach

Pieter Buteneers

Promotoren: prof. dr. ir. B. Schrauwen, prof. dr. P. Boon
Proefschrift ingediend tot het behalen van de graad van
Doctor in de Ingenieurswetenschappen

Vakgroep Elektronica en Informatiesystemen
Voorzitter: prof. dr. ir. J. Van Campenhout
Faculteit Ingenieurswetenschappen en Architectuur
Academiejaar 2012 - 2013

UNIVERSITEIT
GENT

ISBN 978-90-8578-559-0
NUR 981, 984
Wettelijk depot: D/2012/10.500/85

# Dankwoord

*"Het doen van (een promotie-)onderzoek is te vergelijken met het maken van een wielertocht in de bergen: je vraagt je geregeld af waarom je er aan begonnen bent, maar het behalen van resultaat/bereiken van de top geeft voldoening."*

Gerben Duns
Rijksuniversiteit Groningen (2011)

Iedereen die ooit een doctoraat heeft ondernomen en zich aan een fietstocht in de bergen heeft gewaagd, zal deze stelling waarschijnlijk niet tegenspreken en zonder de steun van de vele mensen rondom mij had mijn doctoraat waarschijnlijk niet dit resultaat opgeleverd.

Ik wil in de eerste plaats mijn promotoren prof. Benjamin Schrauwen en prof. Paul Boon bedanken dat ik me 4 jaar lang mocht verdiepen in dit interessante onderwerp. Benjamin heeft me altijd overweldigd met zijn goede ideeën en innovatieve inbreng. Prof. Boon langs zijn kant heeft me laten kennismaken met het menselijke verhaal achter epilepsie. Ook mijn juryleden en in het bijzonder prof. Johan Arends hebben een duidelijk steentje bijgedragen aan het tot stand komen van dit werk.

Uiteraard wil ik ook mijn collega's uit het Reservoir Lab bedanken, niet alleen voor hun technische inbreng maar ook voor de goede sfeer die ze hebben gecreëerd. Dankzij mijn bureaugenoten, Michiel

en Tim, heb ik waar nodig, mijn frustratie kunnen afreageren en elke dag opnieuw kunnen genieten van de nodige afleiding in de vorm van elkaars, soms flauwe moppen. Prof. Joni Dambre en vooral David ben ik altijd dankbaar om hun wetenschappelijk advies. Ze hebben ook mijn werk samen met onze vakgroepvoorzitter prof. Jan Van Campenhout grondig nagelezen en taalkundig telkens weer opnieuw opgekrikt. Daarbuiten denk ik dan aan Francis die mij de weg naar het Reservoir Lab heeft gewezen, aan de jongere collega's Ken en Pieter-Jan met wie ik samen publicaties geschreven heb, maar ook aan Philemon, Sander (clusterman en reddende engel), Aäron, Jonas en tot voor kort Fionntán. De inbreng van de collega's uit het LCEN lab in het UZ Gent heeft me ook vele malen een grote stap verder geholpen en dan denk ik vooral aan Evelien, Bregt, prof. Kristl Vonck, prof. Robrecht Raedt en tot voor kort ook Tine voor het nalezen van mijn werk en het verzamelen van de nodige data om mijn experimenten te doen. Ook de collega's uit MEDISIP, en dan vooral Pieter en tot voor kort Hans, ben ik zeer veel dank verschuldigd.

Van alle thesisstudenten die ik heb begeleid is er één aan wie ik uitdrukkelijk dank verschuldigd ben: Bram. Samen met hem, Paul en Björn vormen we het team achter Pols Healthcare. Ik wil hun dus bedanken om dit uitdagend project samen met mij uit te bouwen.

Natuurlijk ben ik ook mijn vrienden zeer veel dank verschuldigd. Hierbij denk ik vooral aan Geroen en Erik met wie ik samen als drie (on)wijzen vanuit het verre oosten naar Gent getrokken ben, aan het begin van onze hogere opleiding. Mijn klim/ski-vrienden en dan vooral Wim, Paul, Laure, Karen en Tim (trouwens ook mijn bureaugenoot) hebben mij over de jaren heen van de nodige sportieve ontspanning en (over the top) humor voorzien. Daarbuiten wil ik ook Gert, Gunther, Benjamin, Ine, Tim (carbon), Dieter, Roel en alle anderen, die ik hier nu even vergeet te vernoemen, bedanken voor hun vriendschap.

Je familie maakt je letterlijk en figuurlijk tot wie je bent. Daarom wil ik vooral mijn ouders bedanken voor wie ze zijn, de genen die ik van hun heb mogen erven en de kansen die ze mij hebben gegeven. Ook mijn andere familieleden en zeker Oma, Ernest, Tante Anny, Peter, Frederik, Wouter en Pieter (roestbak) mogen hier niet ontbreken.

Als laatste maar zeker niet de minste, wil ik mijn vriendin en levensgezellin Carolien bedanken voor haar steun, geduld en speelse inbreng in mijn leven. Zonder jou was ik deze berg nooit opgeraakt!

Pieter Buteneers
Gent, 1 december 2012

Dit werk werd ondersteund door het Instituut voor de Aanmoediging van Innovatie door Wetenschap en Technologie Vlaanderen (IWT Vlaanderen).

*This work was supported by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT Vlaanderen).*

# Examencommissie

Prof. Hendrik Van Landeghem, voorzitter
Vakgroep ELIS
Faculteit Ingenieurswetenschappen en Architectuur
Universiteit Gent

Prof. Joni Dambre, secretaris
Vakgroep ELIS
Faculteit Ingenieurswetenschappen en Architectuur
Universiteit Gent

Prof. Benjamin Schrauwen, promotor
Vakgroep ELIS
Faculteit Ingenieurswetenschappen en Architectuur
Universiteit Gent

Prof. Paul Boon, co-promotor
Vakgroep Interne Geneeskunde
Faculteit Geneeskunde
Universiteit Gent

Prof. Johan Arends
Vakgroep Electrical Engineering, Signal Processing Systems
Faculteit Electrical Engineering
Technische Universiteit Eindhoven

Prof. Tom Dhaene
Vakgroep INTEC
Faculteit Ingenieurswetenschappen en Architectuur
Universiteit Gent

Prof. Geraldine Heynderickx
Vakgroep LCT
Faculteit Ingenieurswetenschappen en Architectuur
Universiteit Gent

Prof. Ivan Osorio
Vakgroep Neurology
Faculteit Medical Center
University of Kansas

Prof. Andreas Schulze-Bonhage
Vakgroep Neurotechnology
Faculteit University Medical Center Freiburg
University of Freiburg

Prof. Sabine Van Huffel
Vakgroep ESAT
Faculteit Ingenieurswetenschappen
Katholieke Universiteit Leuven

Eerste (interne) verdediging: 21 november 2012, 16h00
Openbare verdediging: 18 december 2012, 15h00

# Samenvatting

## Epilepsie

Epilepsie is een neurologische aandoening van de hersenen die voorkomt bij ongeveer 1% van de wereldbevolking. Ze wordt gekenmerkt door terugkerende epileptische aanvallen en de diagnose wordt gesteld vanaf er zich meer dan één niet uitgelokte epileptische aanval zich heeft voorgedaan. Hoewel er al jaren onderzoek gedaan wordt naar epilepsie, kan men de ziekte nog steeds niet genezen. Onderzoekers overal ter wereld zijn continu op zoek naar manieren om epilepsie beter te begrijpen en doeltreffender te behandelen. Ongeveer twee op drie patienten kan aanvalsvrij worden door middel van medicatie. Bij deze patiënten worden de aanvallen echter vaak vervangen door de bijwerkingen van de medicatie zoals misselijkheid, duizeligheid, een wazig gevoel en ga zo maar door.

In ongeveer één op drie patienten kan medicatie het aantal aanvallen niet of nauwelijks verminderen zonder dat de bijwerkingen de patiënt verhinderen te functioneren. Enkel een fractie van deze patiënten kan geholpen worden door chirurgisch het verantwoordelijke gebied in de hersenen te verwijderen. Voor de andere patiënten is het onmogelijk éénenkel en redundant gebied te vinden in de hersenen. Deze patiënten moeten noodgedwongen leven met steeds weer kerende aanvallen die op elk moment van de dag kunnen optreden.

## Aanvalsdetectie

Voor onderzoekers, patiëten en zorgverleners is epileptische aanvalsdetectie een zeer gegeerd hulpmiddel. Het laat toe om zorgverleners te waarschuwen als een aanval zich voordoet zodat die hem of haar kan bijstaan en het zou toelaten om snelwerkende medicatie of neurostimulatie toe te passen om de aanval te onderdrukken op het moment dat hij optreedt. Deze toepassingen vereisen wel dat de aanvalsdetector een zeer korte detectievertraging heeft, bijna geen aanvallen mist en zo weinig mogelijk vals-positieven detecteeert.

Het electro-encefalogram (EEG) is het meest gebruikte signaal voor het detecteren van aanvallen en wordt ook in dit werk gebruikt. Het meet de elektrische activiteit in de hersenen en wordt toegepast om de diagnose van epilepsie consistent te kunnen stellen.

## Reservoir computing

De meeste aanvalsdetectiealgoritmen uit de literatuur zijn gebaseerd op heuristiche algorithmen die manueel gebouwd werden. Machine learning daarentegen is een tak van de wetenschap waar men tracht algorithmen te bouwen die autonoom een model kunnen leren van bijvoorbeeld een epileptische aanval. Het werd reeds vaak aangetoond dat veel taken beter opgelost kunnen worden met machine learning dan met algorithmen waarbij ontwikkelaars hun interpretatie van het model proberen implementeren.

Er is sprake van zeer veel verschillende machine learning-technieken in de literatuur. De techniek die in dit werk wordt gebruikt is reservoir computing. Ze is gebaseerd op een sterk vereenvoudigd model van de hersenen en gebruikt een willekeuring aangemaakt recurrent neuraal netwerk van artificiële neuronen, het reservoir genaamd. Dit netwerk wordt gevoed met een invoersequentie die door het netwerk gespiegeld wordt op een hoger-dimensionale ruimte. Om dit systeem te trainen wordt enkel een lineaire uitleeslaag getraind op basis van de toestanden van het netwerk. De verbindingen van de invoer naar het reservoir en de verbindingen tussen de neuronen onderling blijven bij dit proces ongewijzigd. Deze aanpak zorgt ervoor dat de tijd die nodig is om het netwerk te trainen binnen de perken blijft zonder dat

er ingeboet moet worden op de prestatie en dat voor een groot aantal taken.

## Leeralgoritmen

Omdat epileptische aanvallen maar zelden voorkomen, bestaan de datasets die in dit werk gebruikt worden vaak uit duizenden uren aan data. Als voor elke seconde aan data één datapunt wordt toegewezen, resulteert dit in een dataset van miljoenen datapunten. Voor een machine learning toepassing is dit zeer groot. Om toch met datasets van deze grootte te kunnen werken, worden in dit werk verschillende algotihmen voorgesteld die toelaten om het systeem te trainen in een grootteorde van minuten in plaats van uren. Elk van deze algoritmen werd aangepast aan de specifieke karakteristieken van epilepsiedatasets zoals het beperkt aantal aanvallen en da variabiliteit van het EEG. Het algoritme dat de beste prestatie levert, schaalt de invloed van elk voorbeeld uit de trainingset aan de hand van zijn relevantie om een goede aanvalsdetector te bouwen.

## Aanvalsdetectie bij diermodellen

Diermodellen zijn nog steeds de enige optie voor epilepsieonderzoek. Om het effect van anti-epileptische behandeling bij deze dieren te onderzoeken worden vaak vele weken aan EEG-data opgenomen. Het markeren van de aanvallen in deze data is dus een zeer tijdrovevende bezigheid voor de onderzoekers.

In recent onderzoek wordt vaak het effect van een zogenaamde closed-loop-behandeling onderzocht. Hier wordt de anti-epileptische behandeling, zoals snel werkende medicatie of neurostimulatie, toegepast van zodra er een epileptische aanval optreedt. Aangezien bestaande aanvalsdetectiealgoritmen vaak niet voldoende betrouwbaar zijn, moet er soms dag en nacht, en zo lang het onderzoek loopt, een onderzoeker aanwezig zijn om de behandeling toe te dienen bij het begin van de aanval.

Om onderzoek in deze domeinen te faciliteren, wordt een nieuw aanvalsdetectiealgoritme voorgesteld dat beter presteert dan de beste systemen uit de literatuur en dat een prestatie heeft die in zekere

zin vergelijkbaar is met die van getrainde onderzoekers. De detectievertraging en het aantal gemiste aanvallen van dit systeem kan gereduceerd worden ten koste van het aantal valspositieve detecties en omgekeerd. Dit laat toe dat epilepsieonderzoekers een aanvalsdetectiemodel kunnen gebruiken dat aangepast is aan hun noden en dat ongeacht de aanpassing toelaat om simultaan aanvallen te detecteren en het EEG te annoteren.

Als een zeer hoge accuraatheid vereist is voor het markeren van epileptische aanvallen was dit tot nog toe niet mogelijk met de huidige technieken. De aanvalsdetector uit dit werk laat toe om dit te bereiken en reduceert de werklast van de onderzoeker met ongeveer 90%. Om de werklast nog verder te verminderen werd een leerstrategie voorgesteld die de tijd die nodig is om de trainingset samen te stellen drastisch vermindert.

## Humane aanvalsdetectie

Voor humane aanvalsdetectie wordt een patiëntspecifiek aanvalsdetectiesysteem voorgesteld dat vergelijkbaar presteert met de beste methodes uit te literatuur. Gebaseeerd op dit model wordt een algemeen aanvalsdetectiemodel voorgesteld dat niet patiëntafhankelijk is. Dit model presteert beter dan vergelijkbare modellen uit de literatuur maar evenaart de prestatie van het patiëntspecifiek model niet.

Twee leerstrategieën werden voorgesteld die toelaten om het algemeen aanvalsdetectiemodel te verbeteren zodat het de prestatie van het patiëntspecifiek model kan bereiken. Het is voor deze strategieën niet nodig om input te vragen van een ervaren neuroloog, maar ze kunnen worden geïmplementeerd als eenvoudige drukken op een knop. De eerste strategie vereist alleen dat de gebruiker op een knop drukt als er een vals-positieve gedetecteerd wordt en slaagt erin om de prestatie van het patiëntspecifiek model te bereiken voor 70% van de patiënten. Als de gebruiker ook op een knop drukt als er een aanval werd gemist, bereikt het systeem de prestatie van het patiëntspecifiek model bij meer dan 90% van de patiënten. Dit gaat echter wel gepaard met een licht verhoogd aantal valspositieven, maar deze kunnen worden gereduceerd door een van beide leerstrategieën toe te passen over een langere periode.

# Summary


## Epilepsy

Epilepsy is a neurological disorder of the brain that occurs in about 1% of the world's population. It is characterized by recurring epileptic seizures and can by diagnosed if more than one unprovoked seizure occurs. Although epilepsy has been a research topic for many years, the perfect cure has not been found. Researchers around the world are constantly looking for new and better ways to understand and treat epilepsy. About two out of three patients can become seizure free using anti-epileptic drugs. For these patients the seizures are often replaced with unwanted side effects such as nausea, dizziness, fuzziness and so on.

In about 1 out of 3 patients, however, the medication does not, or only slightly, reduce the seizure frequency. Only a fraction of these patients can be helped using epilepsy surgery, in which the seizure onset zone is surgically removed. In the rest of the patients, no single and redundant seizure onset zone can be found in the brain. These patients are forced to live with recurring epileptic seizures that can occur at any given point in time.

## Seizure detection

Seizure detection is a much wanted tool for epilepsy researchers, patients and caregivers. It allows caregivers to be alerted to aid the patient while having a seizure and would allow fast-working medica-

tion or neuro-stimulation to be administered to suppress the ongoing seizure. However, these applications require that the seizure detector has very short detection delay, misses almost none of the seizures and has as little false positives as possible.

The electroencephalogram (EEG) is the most commonly used signal to which seizure detection is applied. It measures the electrical activity in the brain and is the most commonly applied tool by neurologists to consistently diagnose epilepsy.

## Reservoir computing

Most epileptic seizure detection algorithms from literature are heuristics that are manually build by engineers. In contrast, machine learning is a research field that tries to build algorithms by allowing a computer to autonomously learn a model from a lot of examples of for instance a seizure. It has been shown that many tasks are better solved using machine learning, rather than by engineers that try to build their interpretation of the model.

There are many machine learning techniques that have been applied to numerous tasks in literature. The machine learning technique used in this work is called reservoir computing. It is based on a simplified model of the brain and uses a randomly created, recurrent network of artificial neurons called a reservoir. This network is fed with an input sequence and maps the input data to a higher dimensional space. To train the system, only a linear readout is trained on the state of the reservoir. The input connections and recurrent connections in the reservoir are left unchanged. This approach dramatically reduces the training time required to train these recurrent neural networks while still attaining state-of-the-art performance in many tasks.

## Learning algorithms

Because seizures are rare events, the datasets used in this work often span thousands of hours. If each second of data is mapped to one data point, this corresponds to millions of data points which is very large from a machine learning perspective. To be able to work with datasets of this size, this work proposes several learning algorithms

that have a training time in the order of minutes as opposed to days. Each of these algorithms has been designed to deal with the special characteristics of epilepsy datasets such as the rareness of seizures and the variability of EEG. The algorithm that performs best scales the influence of each EEG example according to its relevance to build a good working seizure detection model.

## Seizure detection in animal models

Animal models are still the only viable option for research on epilepsy treatment. To evaluate the effect of anti-epileptic treatment, many hours of EEG get recorded. Annotating this data requires many hours of tedious work by experienced encephalographers.

In more recent experiments, the effects of closed-loop epilepsy treatment are investigated. Here anti-epileptic treatment, such as fast-working medication or neuro-stimulation, is applied at the seizure onset. Since most of the current epileptic seizure detection tools are not very reliable, an encephalographer is often present day and night to administer the treatment at the moment of seizure onset.

To aid in these fields of research, a new non animal specific seizure detection model is proposed that outperforms state-of-the-art techniques and has a performance that is comparable with that of encephalographers. The detection delay and the number of missed seizures can be traded of against the number of false positives. This allows epilepsy researchers to select a seizure detection model that optimally fits their needs and allows the tool to be simultaneously applied for on-line seizure detection and seizure marking.

If a very high accuracy is required for seizure marking, the tool can be configured such that the workload of the encephalographers is reduced to up to 90%. To even further reduce the workload, a learning strategy is proposed to bring down the time required to annotate the training data which is needed to build the epileptic seizure detection model.

## Seizure detection in humans

For epileptic seizure detection in humans a patient specific seizure detection model is presented that shows comparable performance to the current state-of-the-art. Based on this model, a non patient specific or general seizure detector is derived that outperforms several techniques from literature. However, this model does not attain the performance of the patient specific model.

Two learning strategies are proposed that allow the general seizure detection model to reach the performance of the patient specific model. These strategies do not require input from an experienced encephalographer, but can be implemented as simple button presses by the user. The first strategy only requires the user to press a button in case of a false positive and allows the model to reach the performance of the patient specific model in 70% of the patients. If the user also indicates when a seizure has occurred, or is ongoing, the performance of the patient specific model is reached in more than 90% of the patients. However, this second approach results in a slightly higher number of false positives. To reduce these, the learning strategies need to be applied over a longer period of time.

# List of Abbreviations

| | |
|---|---|
| AI | Artificial Intelligence |
| AL | Active Learning |
| AOFA | Adapted Osorio-Frei Algorithm |
| ATL | Active Transfer Learning |
| BER | Balanced Error Rate |
| BFS | Backward Feature Selection |
| BRR | Bayesian Relevance Regression |
| BPPT | Back-propagation Through Time |
| DBS | Deep Brain Stimulation |
| EEG | Electro-encephalogram |
| ELM | Extreme Learning Machine |
| EM | Expectation Maximisation |
| FFS | Forward Feature Selection |
| FNPS | False Negatives Per Seizure |
| FPPS | False Positives Per Seizure |
| FS | Feature Selection |
| FS-LS-SVM | Fixed-Size Least-Squares SVM |
| GAERS | Genetic Absence Epilepsy Rat from Strasbourg |
| iEEG | intra-cranial EEG |
| LR | Linear Regression |
| L01 | Zero-One Loss |
| MAP | Maximum A posteriori Probability |

| | |
|---|---|
| ML | Machine Learning |
| OFA | Osorio-Frei Algorithm |
| PSE | Post Status Epilepticus |
| RC | Reservoir Computing |
| RBFS | Regularized Backward Feature Selection |
| RFFS | Regularized Forward Feature Selection |
| RNN | Recurrent Neural Network |
| RR | Ridge Regression |
| SNSR | Seizure to Non-Seizure Ratio |
| SPECT | Single-Photon Emission Computed Tomography |
| SVM | Support Vector Machine |
| SWD | Spike and Wave Discharge |
| TC | Tonic-clonic |
| TL | Transfer Learning |
| TW | Time Window |
| VNS | Vagus Nerve Stimulation |

# Contents

# 1 Introduction

*"Epilepsy is one of the most often misdiagnosed, mistreated, or undertreated conditions..."*

James Firman
President of the U.S. National Council on Aging

## 1.1 Epilepsy

About 1% of the world's population or roughly 100 000 people in Belgium suffer from epilepsy (Witte et al., 2003; de Boer et al., 2008). An epileptic seizure is a sudden, but transient disruption of the electrical activity in the brain. Although not always visible on the outside it can produce very turbulent symptoms. They range from a pounding headache, a lapse in attention or hallucinations to screaming or wild convulsions (Duncan et al., 2006). In healthy people seizures can be triggered by drug abuse, extreme stress, extreme forms of sleep deprivation, etc. However, such seizures are not considered to be symptoms of epilepsy. Epilepsy is a group of diseases that contains all the brain disorders which are characterized by at least two unprovoked seizures.

Epilepsy is a very broad disorder and it can have many causes which can be subdivided into three groups: genetic disorders, provoking circumstances and acquired brain disorders. Although epilepsy can develop because of a single cause, usually the epileptogenisis is

triggered by more than one. A genetic disorder for example can cause epilepsy, but it can also result in a genetic predisposition to develop it in case of a brain injury. Usually it is unclear why a person develops epilepsy at a specific moment and the seizures seem to emerge out of nowhere. In some cases however, the trigger is quite clear, as for example in the case of alcohol abuse, emotional stress, sleep deprivation, a strong fever, etc. Avoiding these triggers does not necessarily mean one will not develop epilepsy, they only increase the probability. Quite often, epilepsy develops because of an obtained brain disorder. It can arise after a brain injury such as a head trauma, a stroke, an intra-cranial haemorrhage, a lack of oxygen during birth, infections such as meningitis or even cancer.

In more than half of the patients, either no cause can be found using thorough medical examinations, or the disorder in the brain is so small that it can not be detected using current techniques (de Boer et al., 2008). Epilepsy can exist and develop at any age, but in about 70% of the cases, the seizures start before the age of 20. A few types are age related and only occur during specific periods in human development such as puberty. In most cases however, the seizures never disappear.

Patients who suffer from uncontrolled epileptic seizures are limited in their independence. They have a significantly higher probability of burns, fractures or other injuries and even sudden death. In many countries they are not allowed to drive and not that many employers want to hire them. Not knowing when a seizure occurs or will occur is a significant burden for the patients and their caregivers.

## 1.2 Epileptic seizures

Epileptic seizures can be subdivided into two main groups: partial seizures and generalized seizures (Angeles, 1981; ILAE, 1989). Figure 1.1 gives a schematic representation of the different seizure types. Partial seizures start in a certain part of the brain cortex and manifest themselves in the corresponding body region or function. Seizures that for example originate in the temporal lobe, the part of the brain

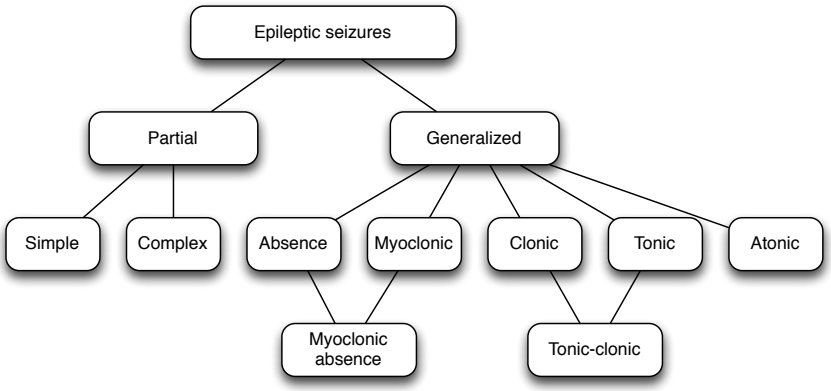


**Figure 1.1:** A schematic representation of the different types of epileptic seizures.

that processes emotions and short-term memory, often result in an aura. During such an aura patients can experience a hallucination of a certain taste or smell, or they may experience certain unexpected feelings like fear, sadness or even euphoria. Most commonly these seizures are caused by an acquired brain disorder. Simple partial seizures do not result in a loss of consciousness. Although they can not control their symptoms, patients are fully aware they are having a seizure. In many situations this type of seizure is harmless for the patients since they can stop their activities until the seizure has passed. Complex partial seizures on the other hand coincide with a reduced level of consciousness. Although patients who are having a complex partial seizure can perform seemingly normal activities, they are unaware of their actions. This is an increased risk for the patient since the seizure is often not recognized as such by those witnessing the event. Partial seizures may spread to involve other regions of the brain and possibly even become secondarily generalized.

Generalized seizures seem to encompass the entire cerebral cortex right from the start and therefore always result in a loss of consciousness. In many cases these seizures only have a genetic cause. They are often subdivided in five groups based on their external characteristics.

- Absence seizure can be recognized by a temporary loss of at-

tention. The patient stops his activity for a short period after which he often resumes his activity as if no time has passed. These seizures exist in adults, but they are most common in young children below the age of 12. They usually last less than a minute and are generally easily treated with medication. However, they are very often a prelude to a more severe type of epilepsy later in life.

- Myoclonic seizures are short jerks of a group of muscles. They are very similar to the jerks people have when falling asleep and are most common around puberty. Patients usually recover very fast after the seizure, which by itself lasts only one or 2 seconds.

- Clonic seizures are successive full body jerks. By themselves they are quite rare and usually follow a tonic phase.

- Tonic seizures are characterized by a tonic contraction of all the muscles. When they are followed by a clonic phase they are referred to as tonic-clonic seizures. This type of seizure is the most common type of generalized seizures in adults. These seizures last between one and five minutes during which patients make wild movements and can even wet themselves. After the seizure the patients are often confused and tired, and it can take several hours before they have fully recovered.

- Lastly there are the atonic seizures that coincide with a total loss of muscle tone. After this type of seizure people usually get up and are able to continue their activities. Atonic seizures are therefore very similar to fainting.

Although tonic-clonic seizures are the most common combination, other combinations may occur such as myoclonic absences.

If an epileptic seizure lasts longer than half an hour and possibly indefinitely, it is referred to as a status epilepticus (Lowenstein and Alldredge, 1998). This can occur with any type of seizures but they are most common and life threatening in the case of a tonic-clonic seizure. If the seizure is not suppressed in time, serious brain and hart damage may occur because of the highly irregular respiration. In case of a status epilepticus the emergency services should be alerted

and if possible a strong anti-convulsant, such as Valium, should be administered.

Because epileptic seizures can manifest themselves in many ways they are often confused with non-epileptic phenomena. These non-epileptic seizures can have a medical origin such as fainting, cardiac arrhythmia, migraine, hyperventilation, transient ischemic attacks, etc. They can also have a psychological origin (Benbadis and Allen Hauser, 2000). These psychogenic non-epileptic seizures or pseudoseizures can be caused by emotional stress. Although they are very difficult to diagnose, they can be successfully treated with psychotherapy. To rule out whether the seizures have a non-epileptic origin or not, the electroencephalogram (EEG) is used.

## 1.3 Treatment

An average human brain contains about 100 billion neurons (Herculano-Houzel, 2009). Each of these neurons is connected to about 1000 other neurons. This means that there are 100 trillion connections in the brain (Drachman, 2005). Although in most cases the exact causes for epilepsy are still unknown, it is accepted in the scientific community that it is caused by problems related to the connections in the brain. Whether it is related to superfluous or missing connections, or that the interconnections are too strong or too weak seems to depend on the specific illness of the patient. Since neurons are very complex cells and there are so many connections in the brain and different causes for epilepsy, there is not one single cure for all epilepsy patients.

About two thirds of the people with epilepsy who have access to healthcare of a westernised standard, can become seizure free with anti-epileptic drugs (de Boer et al., 2008). Finding the right drug or combination of drugs often spans a period of several months or years and it often occurs that in every phase in life a different set of drugs has to be found. Even though, after finding the right drugs the patients are seizure free, they often have to live with the side-effects of the medication: nausea, dizziness, fuzziness, etc (Duncan et al., 2006).

If patients are not be rendered seizure free with the currently developed anti-epileptic drugs, they are said to have refractory epilepsy (de Boer et al., 2008). This is the case for one in three patients. They have the same number of seizures, or only slightly less, while using medication. For some of them, epileptic surgery is another viable solution. In order to qualify, the seizure onset zone must be unambiguously located and must be confined to a single and rather small area in the brain (Quesney et al., 1985). On top of that it must be a redundant area of the brain that is either dysfunctional or replaceable by other areas in the brain. Finding the onset area requires multiple medical examinations. It is often so complex that it has become a research field on its own.

Still one in four epilepsy patients (de Boer et al., 2008), or about 25 000 people in Belgium, are not seizure free with the currently accepted therapies. They are forced to live with their illness or they can try more experimental therapies such as Deep Brain Stimulation (DBS) and Vagus Nerve Stimulation (VNS) to reduce the number of seizures (Vonck et al., 2003; Theodore and Fisher, 2004).

## 1.4 Seizure detection

As mentioned earlier, one of the most significant burdens of epilepsy is not knowing when a seizure will occur. Although it does not appear to be useful to use seizure detection to tell the patient when he is having a seizure, it can help him in many other ways. It can for example aid these patients in offering caregivers a way to know when a seizure occurs. If the detection delay is small enough, caregivers can make sure the patient does not hurt himself. If necessary, an anti-convulsant can be administered to suppress the seizure or the caregiver provide medical care if needed or can call for help (Nijsen et al., 2007).

In research and possible future applications seizure detection can be used to automatically trigger anti-epileptic treatment. This can be in the form of fast-working anti-convulsants or of neurostimulation, such as DBS or VNS (Theodore and Fisher, 2004), to suppress the seizures when they occur. In this set-up a low detection delay is

paramount, since the likelihood of suppressing a seizure seems to decrease with a longer delay before applying the stimulation (Hammond et al., 1992).

If patients are diagnosed with intractable seizures, neurosurgery becomes an option. To find the seizure onset zone, ictal single-photon emission computed tomography (SPECT) is an often used technique (Cysyk et al., 1997). A nurse will inject a radiotracer into the patient as soon as she can distinguish the patient is having a seizure. Because of the increased blood flow to the seizure onset zone, the tracer is absorbed by this brain region. Next the patient is placed under a SPECT scanner so that the seizure onset zone can be visualized. It is obvious that this is a very labour intensive job which could be replaced with an accurate seizure detector with a low detection delay.

Even when selecting the optimal drug or drugs, seizure detection can be useful. Currently the anti-epileptic therapy is often determined based on a patient's description of the severity and number of seizures in comparison to the previous evaluation session. Some patients do not recall having seizures or they can be wrongly evaluated, since they often lose conciousness during a seizure (Hoppe et al., 2007). This can have the effect that patients either get the wrong combination, too much or too little medication. Prescribing too high doses of medication or the wrong combination can have toxic side effects, while too little medication means that the patient still has many epileptic seizures. In some cases, to better determine the optimal set of drugs, patients are hospitalized during a week or so to diagnose which type of epilepsy the patients have. If patients could wear a device that performs accurate automatic seizure detection, physicians would know approximately the number, frequency and duration of the seizures. By correlating this information to the prescribed medication, physicians could converge quicker to a (near) optimal treatment for the patient.

## 1.5 Seizure prediction

Although seizure detection can be very helpful for epilepsy patients, only seizure prediction might lift them from the burden of not knowing when a seizure will occur. Predicting the seizure would give patients time to bring themselves in a safe environment before the seizure occurs and it might allow them to inject themselves with an anti-convulsant much like diabetes patients do with insulin. Even for the applications mentioned above seizure prediction is a welcome bonus on top of seizure detection.

In literature it is argued that reported seizure predictions are simply detections in an earlier stage, i.e., before humans can recognise a seizure (Mormann et al., 2007). Whichever the case, for the patients any indication of when the symptoms will start is welcome. To avoid any discussion on whether seizures can be predicted this work will focus on early seizure detection, detecting seizures with the lowest possible detection delay.

## 1.6 Electroencephalogram

The EEG is a frequently used signal to perform seizure detection. It is a multichannel recording of the electric activity in the brain, where each channel is recorded on a certain brain area. The EEG electrodes can be placed on the scalp or invasively in the brain, in which case it is referred to as intra-cranial EEG (iEEG).

Scalp EEG is most commonly recorded using the international 10-20 system which defines the location of the electrodes as shown in Figure 1.2 (Homan et al., 1987). When visualizing the EEG, each signal or channel represents the brain activity in a certain region in the brain. If mono-polar visualisation is used, the electrical potential difference is measured with one or more reference electrodes. The channel is visually represented by its own symbol, e.g., *F7*. Most commonly bipolar visualisation is used. Here the potential difference is measured between pairs of neighbouring electrodes and it is represented as for example *Fp1-F7*.

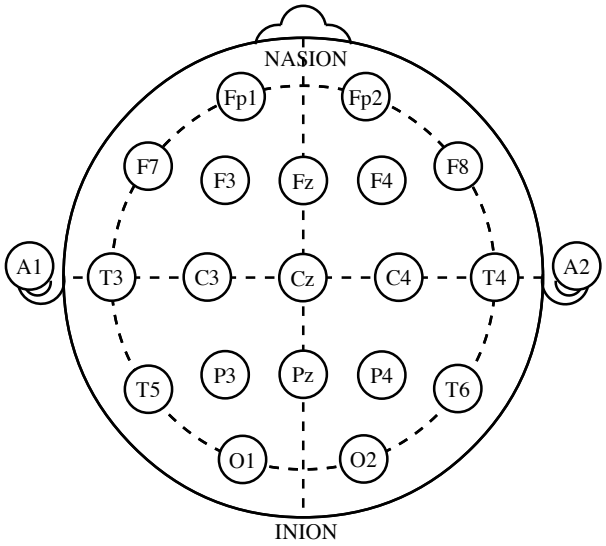


**Figure 1.2:** The international 10-20 system for EEG recording (Homan et al., 1987). The "10" and "20" refer to the fact that the actual distances between adjacent electrodes are either 10% or 20% of the total front-back or right-left distance of the skull. The letters F, T, C, P and O stand for frontal, temporal, central, parietal, and occipital lobes, respectively. Note that there exists no central lobe, the "C" letter is only used for identification purposes. A "z" (zero) refers to an electrode placed on the midline. Even numbers (2,4,6,8) refer to electrode positions on the right hemisphere, whereas odd numbers (1,3,5,7) refer to those on the left hemisphere. In this example the electrodes *A1* and *A2* are two reference electrodes, for which there are a several possible locations. (Figure source: Wikipedia)

There are several physical constraints on what can be recorded using EEG. Since the EEG measures the electrical potential between two electrodes, it can only record the electrical activity parallel to the path connecting them. Activity from neuron interconnections perpendicular to the measurement is not visible on the EEG. The scalp and the cerebrospinal fluid act as electric insulators, with a higher attenua-

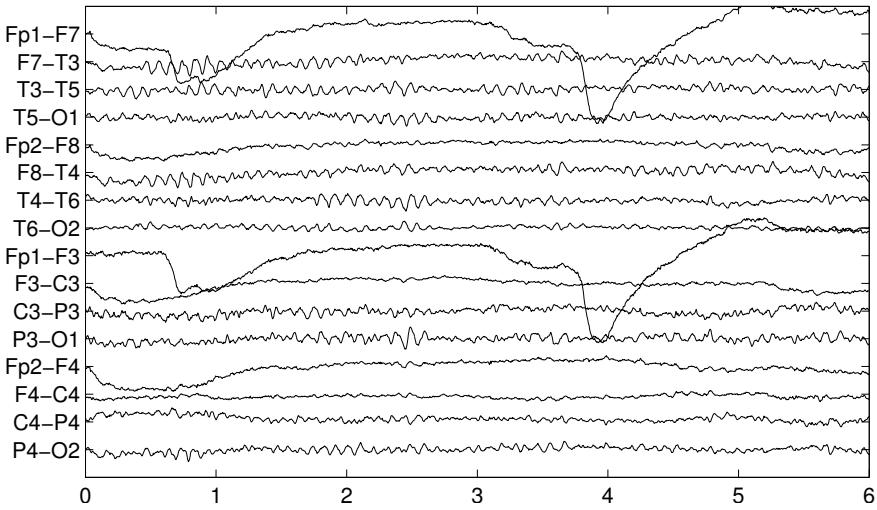


**Figure 1.3:** An example of EEG containing alpha rhythm on the electrodes on the back of the head and eye blinks or eye movement of the left eye at time = 1 s and time = 4 s.

tion of the higher frequency neuron oscillations (Grewal and Gotman, 2005). This constrains the activity that can be recorded to the regions near the scull and the lower frequency ranges. A consequence of these limitations is that seizures with a small and localized activity, somewhere deep inside the brain or perpendicular to the measured electrical field, cannot be observed using scalp EEG.

Electroencephalographers describe EEG activity in terms of its spatial distribution on the scalp (frontal, temporal, parietal, occipital, central, lateral, ...) as well as its dominant frequency component (Pfurtscheller and Lopes da Silva, 1999). These frequency components are grouped in frequency bands of at least 4 Hz. When a subject closes his eyes, the alpha rhythm arises in the EEG. It is characterised by dominant 8 to 12 Hz activity as shown in Figure 1.3 on the posterior channels. The delta rhythm is dominant if the frequency ranges from 0 to 4 Hz, theta if it is between 4 and 8 Hz, beta if it ranges from 12 to 30 Hz and gamma for frequencies higher than 30 Hz. To identify these dominant components the EEG is often visualized with vertical marks

every second. Determining the frequency can then be simplified by counting the number of peaks between each vertical line.

Neuronal activity is not the only activity that is recorded by EEG. There are many physiological artefacts that can be distinguished. Since muscle activity is also triggered by electrical signals, the head muscles have a great influence on the EEG. As shown in Figure 1.3, eye blinks or eye movements are clearly visible as a disruption of the EEG and is most prominent on the channels closest to the eyes. Chewing results in a high frequency activity that disrupts many of the EEG channels as shown in Figure 1.4. Although the main activities that disrupt the EEG are caused by movements of the facial muscles, other activities can disrupt the EEG such as the not uncommon leg shaking shown in Figure 1.5.

### 1.6.1 Seizures on the scalp EEG

Following the onset, an epileptic seizure can be typically recognized as a development of rhythmic activity over several EEG channels. This activity comes from the hyper-synchronous electrical activity of large groups of neurons. It is characterized by an appearance or disappearance of frequency components below 25 Hz for which the exact frequency varies between different patients (Gotman et al., 1981). An example of a generalized seizure is shown in Figure 1.6. The seizure starts at time = 1 s and can be recognised as rhythmic activity on all EEG channels. Typical for this type of seizures is that, as the seizure progresses, the waveforms become rounded with a main frequency around 3 Hz.

Some epilepsy patients have a very irregular EEG signal. The first second of EEG shown in Figure 1.7 is very similar to the generalized seizure shown in Figure 1.6. Although this EEG is abnormal, for this patient it represents the non seizure or inter-ictal EEG and it only disappears whenever there is epileptic activity on the EEG. In this example the seizure starts at time = 2 s, which resembles more the normal EEG of regular patients. However, on the occipital channels a clear rhythmic activity is visible, corresponding to the activity of a partial epileptic seizure.

Each seizure type is represented by a different EEG pattern. The

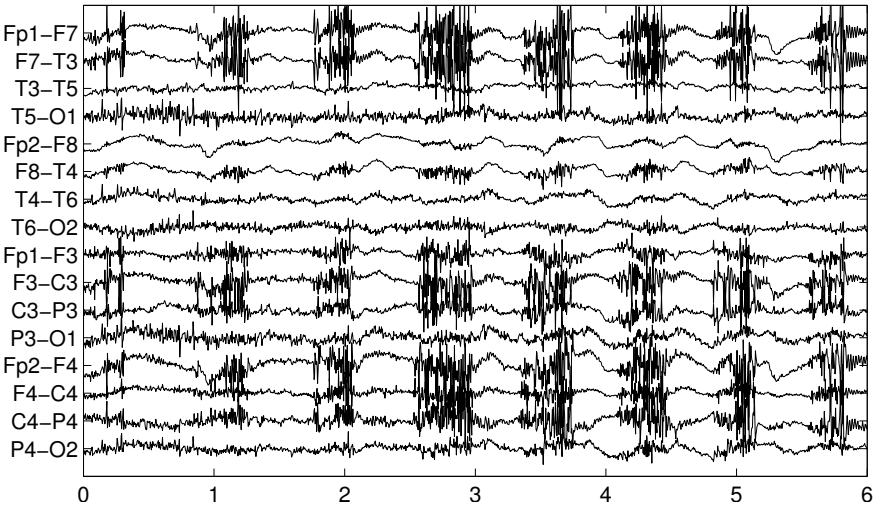


**Figure 1.4:** An example of EEG disrupted by rhythmic activity caused by chewing. This artefact is mainly visible on the electrodes which contain a signal from the central electrodes C3 and C4.

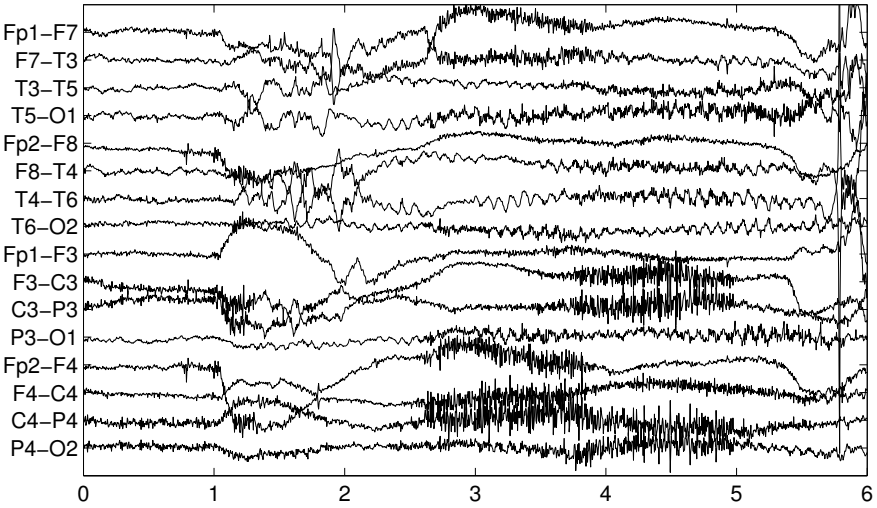


**Figure 1.5:** An example of EEG disrupted by rhythmic activity caused by stress related leg shaking which starts at time $= 1$ s.

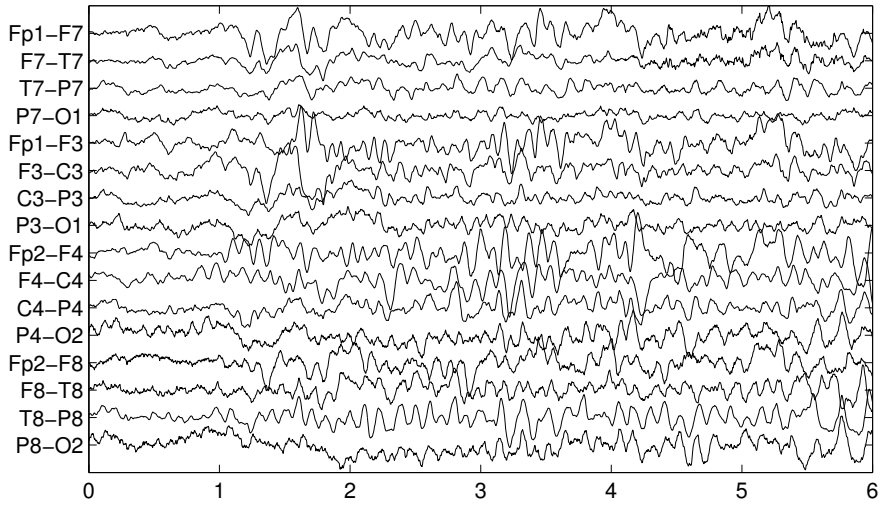


**Figure 1.6:** An example of the onset of a generalized seizure. The seizure starts at time = 1 s. The part of the seizure that is not illustrated, about 35 seconds in length, is characterized by similar rhythmic activity as the last second shown in the figure.

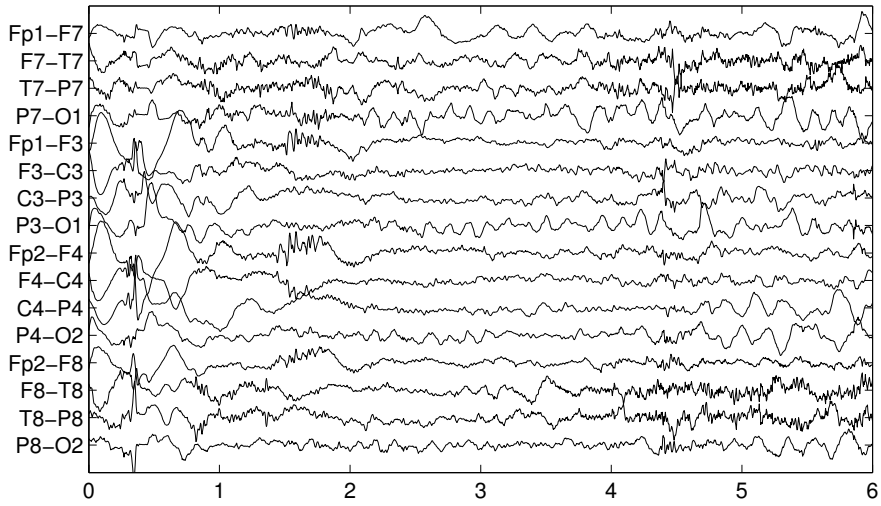


**Figure 1.7:** An example of the onset of a partial seizure. The seizure starts at time = 2 s and is mainly visible on the occipital electrodes.

generalized seizure shown in Figure 1.8, is for example characterised by higher frequency oscillations than generalized seizure shown in Figure 1.6. The latter has a main frequency around 3 Hz whereas the former seizure has a main frequency around 20 Hz. Although there is some similarity within one seizure type, there is a lot of variation between patients and even between seizures of the same patient. In addition, the EEG of one patient during a seizure may closely resemble the signature of abnormal, inter-ical EEG from the same patient as shown in Figure 1.9. These different types of variability make epileptic seizure detection a non trivial task.

### 1.6.2 Seizures on the intra-cranial EEG

Because it is measured on the brain cortex or deep within the brain structures, iEEG measures the electrical activity of a smaller population of neurons and is less sensitive to artefacts. Movement of facial muscles has no influence and the signal strength is not attenuated by the scull. As a consequence, the signal quality and spatial resolution of iEEG is much higher than with scalp EEG.

Therefore, and because the electrodes can be implanted relatively close to the suspected seizure onset zone, it is often used in pre-surgical evaluation. If the iEEG is recorded close to the seizure onset zone, a seizure is visible on the iEEG much earlier, especially in the case of partial seizures. This latency can be as large as several 10s of seconds before the onset is visible on the scalp EEG (Pacia and Ebersole, 1997). At the same time, the higher spatial resolution of iEEG permits the recording of a wider gamut of abnormal, non-seizure activity that is not visible on the scalp EEG (Jirsch et al., 2006; Tao et al., 2005; Urrestarazu et al., 2007). However, this activity can hamper the seizure detection process.

On the iEEG, epileptic seizures manifest themselves as a sudden redistribution of spectral energy on a set of iEEG channels. This change typically consists of an appearance or disappearance of frequency components within the 0-65 Hz band (Grewal and Gotman, 2005). Since there is no attenuation of the skull or the cerebrospinal fluid, this frequency range is broader than with scalp EEG.

In Figure 1.10, an example is shown of a partial seizure on the

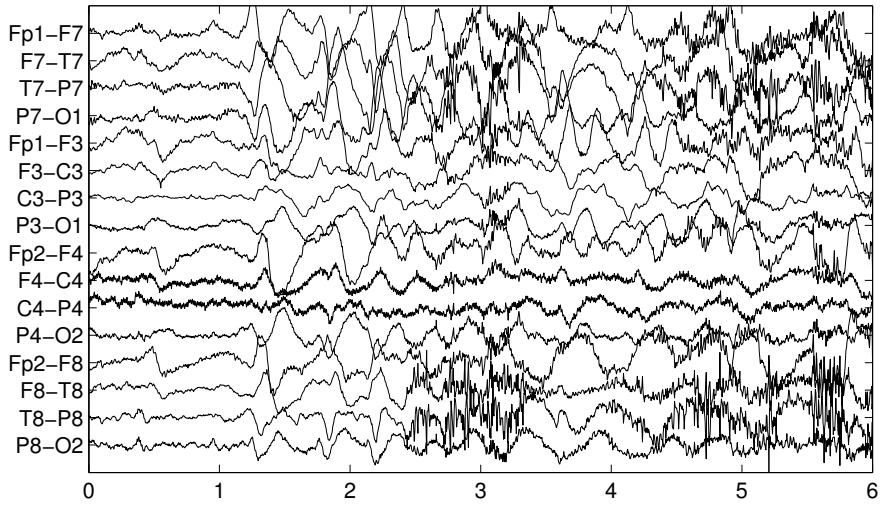


**Figure 1.8:** An example of the onset of a tonic clonic seizure. The seizure starts at time = 1 s. The part of the seizure that is not illustrated, about 60 seconds in length, is characterized by similar rhythmic activity as the last second shown in the figure.

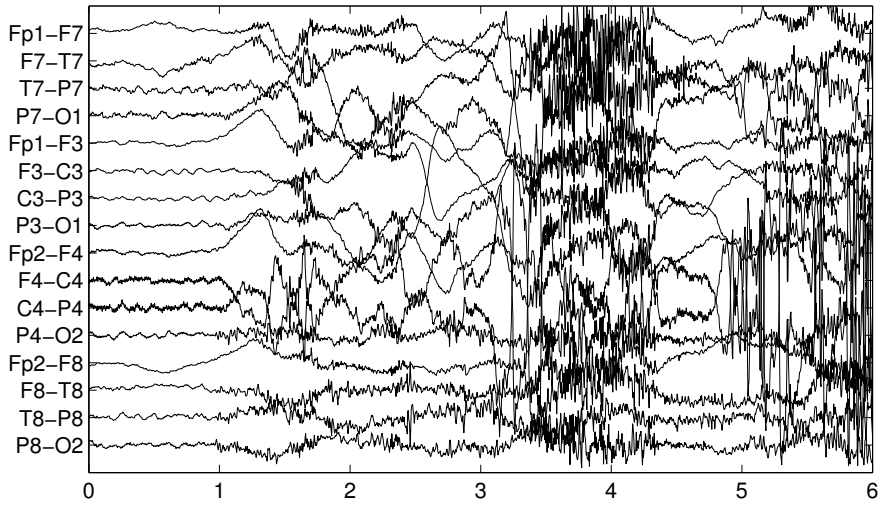


**Figure 1.9:** An example of an inter-ictal epileptic discharge distorted with motion artefacts which start at time = 1 s.

iEEG. The first 3 channels originate from the seizure onset zone. The last 3 channels are from a brain structure that is relatively far away from the seizure onset zone and that is not involved in the seizure. The seizure starts at time = 1 s with rhythmic spiking, mainly on channel *IH4*. Between time = 4 s and time = 9 s high frequency activity is visible on channel *G_A4* with a frequency between 30 and 35 Hz. During this period there is practically no seizure activity visible on channels *IH4* and *IH3*. At time = 9 s the rhythmic activity on channel *G_A4* starts to slightly fade away while it seems to be replaced by spike and wave discharges of about 3 Hz on channels *IH4* and *IH3*.

In Figure 1.11 a different seizure from the same patient is shown. Here the seizure starts at time = 2 s with similar 30 to 35 Hz activity on channel *G_A4* as in the seizure shown on Figure 1.10. Here however the activity is not replaced by spike and wave discharges on channels *IH4* and *IH3*, and the seizure stops at time = 9 s. The seizure in Figure 1.11 is preceded by similar spikes on channel *IH4* as in the beginning of the seizure shown in Figure 1.10. Here, however, these spikes have not been considered as part of the seizure and are marked as inter-ictal EEG. This illustrates that different seizures can be marked differently by encephalographers. How a seizure should be marked is often a subject of debate.

Some seizures are not visible on the EEG. In Figure 1.12 an example is shown of a seizure that can not be visually recognised using the grid electrodes placed on the cortex. These seizures are recognised using video-EEG monitoring.

Similarly to scalp EEG it can also occur that seizures are almost identical to abnormal non-seizure EEG as shown in Figures 1.13 and 1.14. Both figures show similar activity on the first three EEG channels. Usually epileptic inter-ictal activity is characterized by a single short burst. However in the example in Figure 1.13 the seizure consists of shorter bursts than inter-ictal burst shown in Figure 1.14. Then again the total length of the activity is about 25 seconds for the seizure and 10 seconds for the inter-ictal burst.

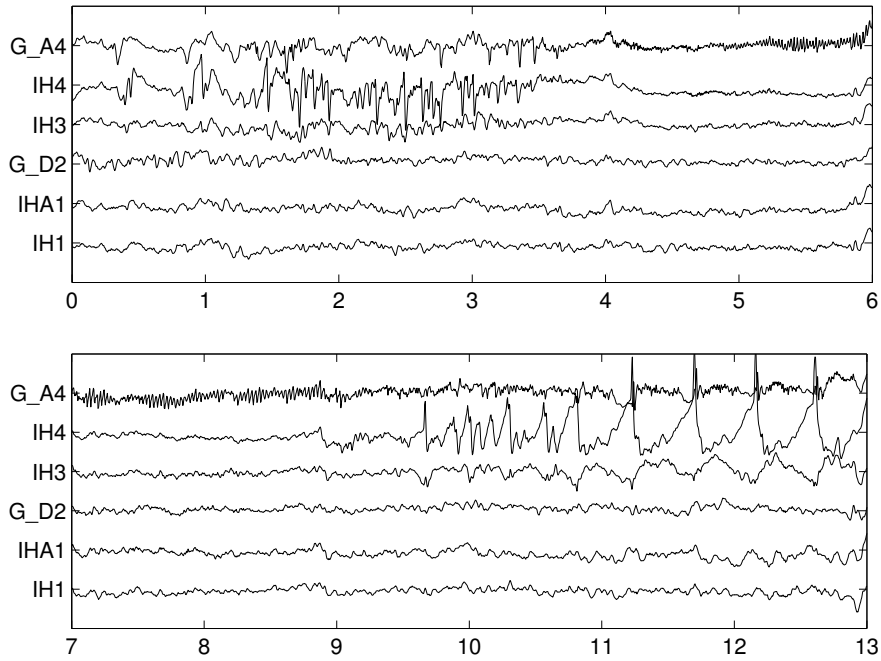


**Figure 1.10:** An example of the onset of a partial seizure on the iEEG which starts at time = 1 s. The not illustrated seizure part, about 15 seconds in length, is characterized by similar rhythmic activity as the last 2 seconds.

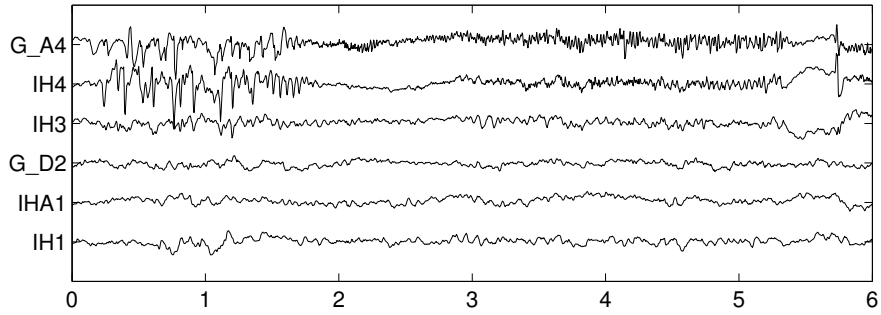


**Figure 1.11:** An example of a different epileptic seizure from the same patient as above. The seizure starts at time = 2 s and stops at time = 9 s (not illustrated).

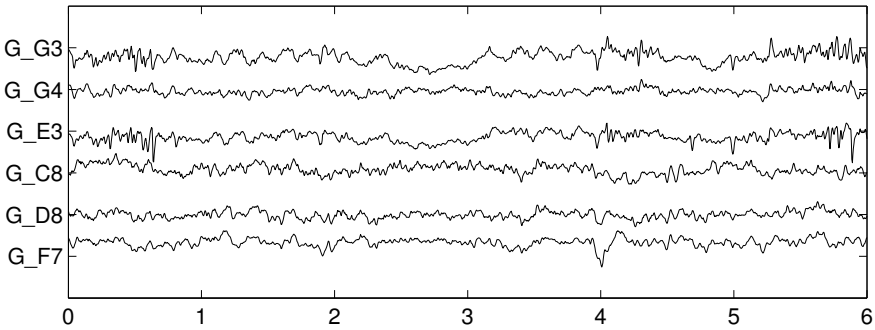


**Figure 1.12:** An example of a seizure that is not visible on the iEEG which starts at time = 2 s. The not illustrated part, about 35 s, shows similar activity.

## 1.7 Quality measures for seizure detection

The gold standard used to compare the different detection methods is the scoring by experienced encephalographers. As mentioned earlier, there can be some inconsistency how the seizures are marked.

Most commonly in literature, seizure detection techniques are evaluated by their detection delay, percentage of missed seizures and the number of false positives per 24 hours. Based on the remarks by Jean Gotman at the International Workshop on Seizure Prediction in 2010[1], a different measure for the false positives will be used in this work. Because not every patient has the same number of seizures per 24 hours and it is more relevant for the above mentioned applications to measure the number of false positives per seizure (FPPS). This is the number of falsely detected seizures divided by the total number of true seizures.

The number of missed seizures is given in false negatives per seizure (FNPS), which is the number of missed detections divided by the total number of true seizures. To compare the detection delays

[1]Personal communication

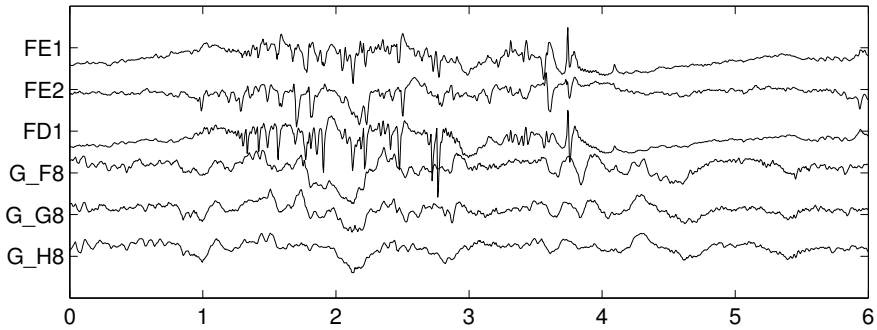


**Figure 1.13:** An example of the onset of a partial seizure on the iEEG which starts at time = 2 s. The not illustrated seizure part, about 20 seconds, is characterized by similar rhythmic activity in bursts.

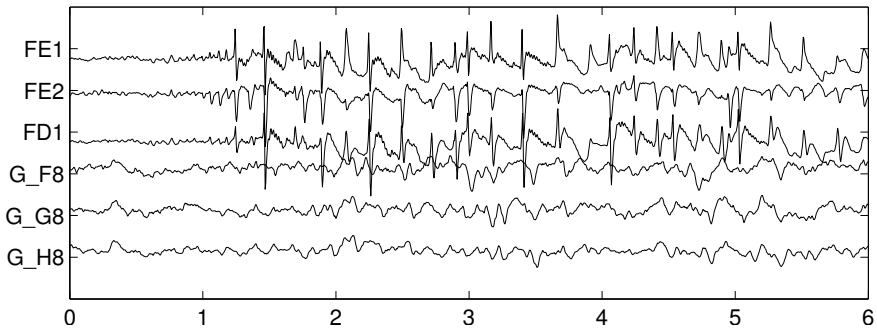


**Figure 1.14:** An example of an inter-ictal burst that is very similar to an epileptic seizure. It starts at time = 1 s and stops at time = 11 s (not illustrated).

of the (on-line) detection methods $\Delta_{delay}$ is measured, the average detection delay in seconds. It is only determined for correct seizure detections and includes the time required to perform preprocessing. As a lower bound, the first inter-ictal sample after the previous seizure is used and as an upper bound, the last marked sample of the to be detected seizure.

## 1.8 Related work

One of the first epileptic seizure detection algorithms was developed by Gotman (1982). The algorithm searches for rhythmic activity with a dominant frequency between 3 and 20 Hz. A seizure is detected if this activity has an amplitude of 3 times larger than normal EEG, is present on at least 2 channels and persists for at least 4 seconds. It is very effective for seizures with a fundamental frequency below 20 Hz. However, it fails to detect seizures with a mixture of frequencies, a low amplitude or a fundamental frequency above 20 Hz. Because of its simplicity it is known to detect many fractions of normal EEG as a seizure, such as sleep spindles, artefact induced bursts and so on. The algorithm has recently been tested by Saab and Gotman (2005) on a dataset containing 126 seizures from 28 patients with a total of 652 hours of EEG. It was able to detect 50% of the seizures with 2.6 FPPS with a median detection delay of 14.3 seconds.

In Osorio et al. (1998) the most frequently cited seizure detection technique was proposed and is designed specifically for iEEG. It first filters the iEEG using a wavelet filter. In practice, this filter compares the shape of the waves in the EEG signal with a shape of high resemblance to a seizure. Next, this signal is rescaled using background EEG. If the resulting signal surpasses a certain threshold, it is considered part of a seizure. For a more detailed explanation we refer to Section 4.3.1 of this work and literature. The algorithm was recently tested in Osorio et al. (2002) on a 70 hour dataset from 14 subjects containing 34 seizures. The algorithm was able to detect 100% of the seizures with a median detection delay of 3.6 s and only 0.2 FPPS.

Since the early work of Gotman and Osorio et al., many seizure detection algorithms have been developed (Tzallas et al., 2012). Table 1.1 compares the performance of the methods relevant to this work. One of these algorithms is the Reveal algorithm developed by Wilson et al. (2004). The technique will be discussed in Section 5.3.2, since it has been compared in Shoeb (2009) on the dataset used in this work. In Wilson et al. (2004) it has been reported to be able to detect 76% of 672 seizures from 426 individuals with in total 1049 hours of EEG. It achieved 2.6 false positives per 24 hours on a 465 hours

**Table 1.1:** The average FPPS, FNPS, detection delay in seconds for the relevant methods from literature. These values can not be compared since the methods have been tested on different datasets. Detection delays marked with (*) represent the median as opposed to the average.

| **Methods** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| Gotman (1982) | 2.6 | 0.5 | 14.3* |
| Osorio et al. (1998) | 0.2 | 0 | 3.6* |
| Wilson et al. (2004) | n/a | 0.24 | n/a |
| Saab and Gotman (2005) | 1.8 | 0.24 | 10* |
| Shoeb (2009) | 0.51 | 0.09 | 4.6 |
| Gardner et al. (2006) | 11 | 0.03 | -7.6 |
| Balakrishnan and Syed (2012) | 1.4 | 0.03 | 7.9 |
| Mirowski et al. (2009) | 0 | 0.29 | -3600 |

dataset of 33 non-epileptic patients. No reports were made about the detection delay. The same dataset was used for training and testing, and epilepsy patients have more abnormal rhythmic, non-seizure activity in the EEG. Therefore, these results are probably not representative for its real life performance as will be shown in Chapter 5.

Saab and Gotman (2005) developed one of the most cited and the first non patient specific seizure detector that was specifically designed to perform seizure onset detection. The algorithm uses features derived from a wavelet decomposition of each EEG channel to detect a seizure. On a test set of 360 hours off EEG from 16 patients containing 69 seizures it was able to detect 76% of the seizures with a median delay of 10 s and 1.8 FPPS.

In Shoeb et al. (2004) and Shoeb (2009) a patient specific seizure detection algorithm was presented. This algorithm is considered to be the current state-of-the-art in patient specific seizure detection and will be discussed in more detail in Section 5.3.3. It was tested on the dataset used in this work which contains 964 hours of EEG and 169

seizures from 23 paediatric patients. With an average detection delay of 4.6 s it only missed 9% of the seizures and achieved 0.51 FPPS. It needs to be trained on about 24 hours of EEG and at least 3 seizures to achieve this performance. Although the training was done on a separate dataset from the testing, there is no mention on how certain training parameters were chosen. These parameters could have been chosen based on the performance on the test set. This might imply that similar performance will not be achieved on other datasets. An attempt at reconstructing these results on a subset of the data was done in Balakrishnan and Syed (2012). Here it only missed 3% of the seizures but 1.4 FPPS were detected and a detection delay of 7.9 s was reported. To achieve these results different training parameters needed to be used because the parameters mentioned in Shoeb (2009) yielded a system unable to detect any seizures.

Since seizures are rare events, gathering training data containing correctly marked seizures is very labour intensive. In Gardner et al. (2006) a technique was presented that can be trained using solely inter-ictal EEG. It was tested on a rather small dataset of 17.5 hours of intra-cranial EEG from 5 patients containing 29 seizures. The inter-ictal EEG was randomly selected from a 200 hour dataset and they made sure it contained no recording artefacts. On this dataset the authors were able to detect the seizures 7.6 s before the marked seizure onset and missed only 3% of the seizures. The false positive rate however was 37 false positives per 24 hours. Since the data is a subset of a real dataset, it is impossible to determine the exact number FPPS. If the same performance would have been achieved on the missing parts of the dataset, roughly 11 FPPS would have been detected.

Another attempt at reducing the work needed to build a training set was presented in Balakrishnan and Syed (2012). It trains the method by Shoeb et al. using active learning (AL) and achieves comparable performance while only requiring 4% of the labelled data. For more details on AL the reader is referred to Sections 2.1 and 4.9.

In literature many techniques have been proposed that can predict epileptic seizures. However almost none of them have been successfully tested on patients in real-world situations and their results are still highly contested in the scientific community (Mormann et al.,

2007). The method which can be considered the current state of the art was published by Mirowski et al. (2009) and was tested on the same iEEG dataset used in this work. For every patient, 18 possible combinations of 6 types of features and 3 types of classifiers were tested and at least one method was able to predict the seizure about 60 minutes before the marked seizure onset. When the best performing method was used for each patient, only 29% of the seizures were missed and no FPPS were detected. However, there was no single method that worked for all patients. The best performing feature-classifier combination worked on 15 of the 21 patients. For the other patients the results were not shown in the paper because of bad performance. Although this is the best performance ever achieved on this dataset, the results still need to be validated in a long term experiment and the authors fail to provide a strategy to select the best feature, classifier and training algorithm for individual patients.

## 1.9 Animal Models

Although ethically disputed (Moore, 1989), animal models are still commonly used for research purposes. For epilepsy they are the only available alternative to evaluate the therapeutic efficacy of anti-epileptic treatment (Dedeurwaerdere, 2005). In order to validate these treatments the number of seizures and their duration needs to be determined. This results in many hours of tedious EEG review and analysis. Automated seizure detection decreases the workload and may also be more reliable and reproducible compared to hours of visual analysis. The advantage of accurate real-time seizure detection is the potential to incorporate this detection into a so called closed-loop system. It allows immediate triggering of an intervention at the time of seizure occurrence such as: fast working anti-epileptic drugs, DBS (Waterschoot et al., 2006; Wyckhuys et al., 2010), VNS (Boon et al., 2001), ...

Most methods extract simple features from the EEG such as the amplitude of the EEG signal (Fanselow et al., 2000), the slope (Westerhuis et al., 1996) or the energy (Van Hese et al., 2003). In a next

phase these feature signals are usually windowed and a simple threshold is applied to determine if the samples within this window are part of a seizure or not. The method in White et al. (2006) is slightly more complicated and combines an amplitude based autocorrelation measure with a spike detector. For more details on these methods we refer to literature. They were recently tested in Buteneers et al. (2010) using different error measures where they were all outperformed by the method for human seizure detection by Osorio et al. (1998) discussed above. This method will be used for comparison in Chapter 4. In Van Hese et al. (2009) a method to detect absence seizures in animal models was proposed that used the specific frequency components of such a seizure and compared this to a background signal. In Buteneers et al. (2010) it detected 3.5 FPPS and missed 11% of the seizures. No detection delay was given, since this method was not designed for on-line seizure detection. For more details on this method we refer to Section 4.3.2 and literature.

In Nandan et al. (2010) a technique to detect epileptic seizures in rats was presented which extracts several features from the EEG and compared several learning algorithms. In their results the authors describe a method that was able to detect seizures about 10 to 15 seconds before the seizure onset. However, this resulted in more than 100 FPPS[2]. An average TC seizure in PSE rats lasts about 1 minute and occurs about every 40 minutes. If one would apply DBS or VNS for the duration of a seizure when it is detected, you would reach near continuous stimulation from about 40 FPPS. For these reasons this method can be considered unsuitable for practical use.

## 1.10 Contributions and structure

The contributions of this work can be grouped into three clusters:

- New algorithms are proposed to train recurrent neural networks using the reservoir computing approach. These algorithms have been designed to have low computational cost and memory

[2]This was not specifically stated in Nandan et al. (2010), but it can be deduced from the results and was confirmed by one of the authors.

requirements for large datasets such as the datasets used for epileptic seizure detection. These algorithms have been compared for epileptic seizure detection on data from animal models.

- For animal models, a non animal specific seizure detection algorithm is proposed which achieves state-of-the-art performance. It has been validated on datasets from two different animal models: the genetic absence epilepsy rats from Stassbourg (GAERS) and the post status epilepticus (PSE) model. Its performance has been shown to be only slightly worse than that of human encephalographers and outperforms all tested methods from literature. It allows for the system to be used as a tool to automatically mark epileptic seizures on the EEG and as an on-line seizure detector for research towards closed loop anti-epileptic treatment.

  To even further reduce the detection delay and the number of missed seizures, a threshold parameter can be lowered. This allows for a faster response time in closed loop experiments at the cost of a few more FPPS. Using this lower threshold, this set-up can also be applied for highly accurate seizure marking, which is possibly better than human performance. It requires encephalographers to only review detected seizures and since seizures are rare events, this significantly reduces the workload. To even further reduce the workload, an active learning strategy has been proposed.

- For human epilepsy patients, a patient specific seizure detection system was proposed that performs comparable to the current state-of-the-art. Based on this set-up, a non patient specific seizure detector was built, which was able to outperform several algorithms from literature. However, the performance of the patient specific seizure detection model was not attained.

  To build a patient specific seizure detector without the need for large amounts of marked data, two learning strategies were proposed. They allow the behaviour of the common seizure detector to be adapted to the patient without requiring experienced

encephalographers. The first strategy only requires the patient and/or caregiver to indicate when a false positive is detected and achieves comparable performance to the patient specific model in 70% of the patients. The second strategy achieves this performance in 90% of the patients, and requires the user to be able to indicate when a seizure was missed. These methods do not need the EEG to be marked by costly encephalographers, but can be implemented with simple button presses.

This thesis is structured as follows. In the next chapter, the basic principles of machine learning are explained in more detail. Reading this chapter will allow the reader to better understand the topics discussed in the following chapters. In Chapter 3, the technical aspects of the newly developed training techniques for reservoir computing (RC) are discussed. These techniques are validated in Chapter 4, where the seizure detector for animal models is introduced. In Chapter 5, the seizure detection model for human EEG is introduced. The last chapter concludes my work and gives an overview of the possible directions for further research. Throughout this work a clear distinction will be made to separate the more technical sections which can be skipped on a first reading.

## 1.11 List of publications

### Journal publications

1. Buteneers, P., Caluwaerts, K., Verstraeten, D., and Schrauwen, B. (2012). Optimized parameter search for large datasets of the regularization parameter and feature selection for ridge regression. Neural Processing Letters. (under revision).

2. Buteneers, P., Verstraeten, D., Nieuwenhuyse, B., Stroobandt, D., Raedt, R., Vonck, K., Boon, P., and Schrauwen, B. (2012). Real-time detection of epileptic seizures in animal models using reservoir computing. Epilepsy Research. (in press)

3. Verstraeten, D., Schrauwen, B., Dieleman, S., Brakel, P., Buteneers, P., and Pecevski, D. (2011). Oger: Modular learning architectures for large-scale sequential processing. (in press).

4. Buteneers, P., Verstraeten, D., van Mierlo, P., Wyckhuys, T., Staelens, S., Stroobandt, D., and Schrauwen, B. (2010). Automatic Detection of Epileptic Seizures on Intra-cranial EEG from Rats using Reservoir Computing. AI in Medicine, 53:215-223.

## Conference publications

1. Pieter Buteneers, David Verstraeten, Robrecht Raedt, Dirk Stroobandt, Kristl Vonck, Paul Boon and Benjamin Schrauwen. Real-time detection of epileptic seizures in animal models using reservoir computing. HIVE Worshop - Berlin (2012)

2. Pieter-Jan Kindermans, David Verstraeten, Pieter Buteneers and Benjamin Schrauwen. How do you like your P300 speller : adaptive, accurate and simple? 5th international brain-computer interface conference, Proceedings, pp. 4 (2011)

3. Pieter-Jan Kindermans, Pieter Buteneers, David Verstraeten and Benjamin Schrauwen. An uncued brain-computer interface using reservoir computing Workshop : machine learning for assistive technologies, Proceedings, pp. 8 (2010)

4. Pieter Buteneers, Benjamin Schrauwen, David Verstraeten and Dirk Stroobandt. Real-time epileptic seizure detection using reservoir computing Seizure Prediction, 4th International workshop, Abstracts, pp. (2009)

5. Pieter Buteneers, Benjamin Schrauwen, David Verstraeten and Dirk Stroobandt. Real-time epileptic seizure detection on intracranial rat data using reservoir computing Lecture notes in computer science, Vol. 5506, pp. 56-63 (2009)

6. Pieter Buteneers, Benjamin Schrauwen, David Verstraeten and Dirk Stroobandt. Epileptic seizure detection using Reservoir Computing Proceedings of the 19th Annual Workshop on Circuits, Systems and Signal Processing, pp. 1-4 (CD-ROM) (2008)

# 2 Machine learning and reservoir computing

Although this chapter covers several technical aspects of the learning algorithm used in this work: reservoir computing (RC) and machine learning (ML) in general, an attempt has been made to explain the most relevant parts in such a way that they can be understood by people with a basic scientific background. Sections that are not necessary for understanding the conclusions from this work and require a more mathematical insight in the matter, will be clearly marked and can be skipped.

To cover all aspects of ML would lead us far out of the scope of this work. Instead only the subjects relevant to grasp the rest of this work will be discussed. For more detailed information, we refer to literature (Bishop, 2006).

## 2.1 Machine learning

ML, or artificial intelligence as it is more commonly known, is often hyped in science fiction. However, the ML systems that exist in our daily life are far less developed than one would hope. Nevertheless, most people think they have never come in contact with any form of ML but less is true. Many things we use daily like Google, smartphones, washing machines, the electrical grid and so forth wouldn't function like we know them to without ML.

In the ML domain, machines are programmed to learn to solve

certain tasks rather than having to explicitly program every step towards the solution. One of these learning algorithms is RC which is discussed below and will be used throughout this work. In practice, ML algorithms are most commonly used for classification such as identifying faulty products on a production line. The system is presented with a certain input from which it is supposed to decide which class can be correlated to the input. Instead of manually defining the boundaries of a certain input, algorithms are used that can learn these boundaries.

ML tries to model a real process using a set of inputs. Each ML technique makes a set of assumptions about the model. These assumptions or prior distributions of the model can be very simple, for example: if it was sunny yesterday, it is more likely to be a bright day. Making assumptions that do not fit the reality will of course result in a model that does not resemble the reality. If a weather forecast model is built upon the prior assumption that today's weather is only dependant on last years weather, the forecast will be off most of the time. The art in ML is to use the smallest number of hand tuned priors and use learning techniques to find the optimal priors and parameters.

Since ML techniques learn a model, they require data to learn from: the training set. To evaluate the performance, the model is usually applied to a test set. This test set should under no circumstance be used for training or optimizing the training parameters, because if you know the answer, it is easy to find a technique that can answer the question. But this is not a good indication as to whether the model will be able to answer other questions.

Many learning algorithms have been developed and they can be categorised in many ways. One way of doing this, is by looking at the data that was used to train the system. In supervised learning the system is trained on data for which the desired output is known. Unsupervised learning on the other hand covers the domain of learning from data for which the desired output is not known. In each of these clusters many algorithms have been developed, as well as intermediate approaches such as semi-supervised learning.

Supervised and semi-supervised learning have many variants from which two are worth mentioning in this work: active learning (AL) and

transfer learning (Krogh and Vedelsby, 1995; Pan and Yang, 2010). In AL the system is trained on a small dataset. Next the system is evaluated on the data with unknown output and every time the system is uncertain it asks for active input from the user. Much like a child that asks if it is doing its homework right and learns to perform better with some limited input from a parent or teacher. This technique is most commonly used on datasets that are very time-consuming or costly to annotate.

With transfer learning the system is trained on data with some similar properties to the data for which it will be used. When it is applied on the task at hand the system will adapt to it. This adaptation can be done unsupervised or supervised. Transfer learning is commonly used in dictation software. It can be seen as a system that has been trained on the data of many different people in order to perform well on people on which it has not been trained.

## 2.2 Linear regression

One of the most basic and most commonly used techniques in ML is linear regression. It thanks this privileged position not only to its simplicity but also to its effectiveness for many tasks. It assumes that there is a linear relation between the input and the output. Although this is almost never exactly true, it is often a very good approximation of the reality.

We will demonstrate this with an example. Plotting the logarithm of the heart rate of mammals as a function of their life expectancy, as illustrated in Figure 2.1, shows that there is an almost linear relation between them (Levine, 1997). The fact that humans appear to be an exception, is mainly because we are now well nourished and have good healthcare. In the early $20^{th}$ century, our average life expectancy was only 31 years (Riley, 2001).

Linear regression is a technique to estimate weights that determine this linear relation. If there is a perfect linear relation between a heart rate $y$ and a given life expectancy $x$, the heart rate can be estimated

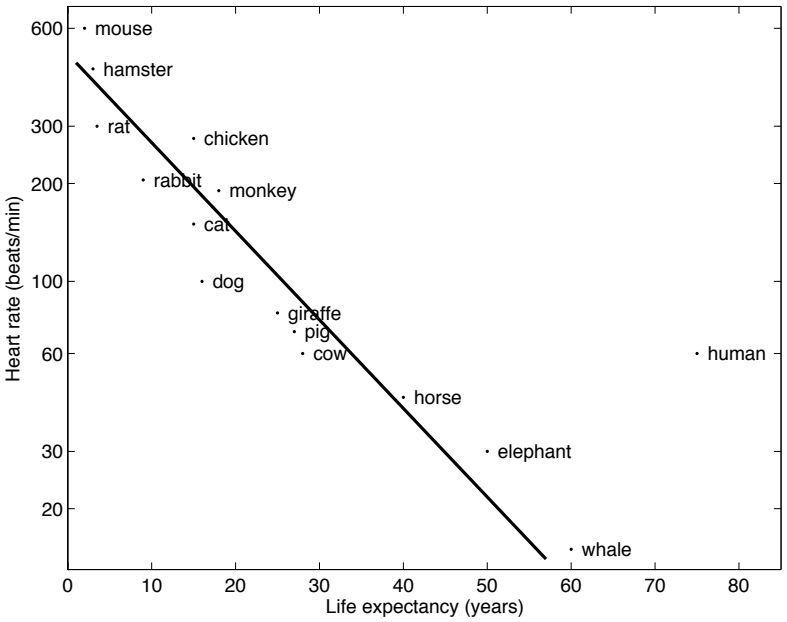


**Figure 2.1:** The heart rate as a function of the life expectancy in mammals.

as follows:

$$y = w_1 x + w_0,$$

where $w_1$ determines the slope of the line and $w_0$ represents the vertical shift of this line, the $y$-intercept or so called bias. However, as in the example of Figure 2.1 there usually is only a near linear relation. Linear regression tries to minimize the quadratic error which can be determined as follows:

$$f_{loss}(x, y) = (y - w_1 x - w_0)^2, \tag{2.1}$$

where $f_{loss}$ is the error function or loss function we try to minimize. In the example from Figure 2.1 there is only one feature and that is the life expectancy. For the reader interested in the mathematical details we can generalize this in the following way. In a more realistic

setting, if there are $N$ features, the loss function becomes:

$$f_{loss}(x, y) = (y - \sum_{i=0}^{N} w_i x_i)^2,$$

where $x_i$ represents feature $i$ of the $N$ features to which $x_0 = 1$ is added for mathematical simplicity. In matrix notation, for all possible examples in the training set, this becomes:

$$f_{loss} = ||\mathbf{Y} - \mathbf{X}\mathbf{W}||^2,$$

where $\mathbf{X}$ is the input matrix with size $M \times N$, with $M$ the number of data points in the training set, and $\mathbf{Y}$ contains the desired output and has size $M \times P$, with $P$ the number of outputs. Minimizing this loss to find the optimal weights $\mathbf{W}_{opt}$ yields the following solution:

$$\begin{aligned} \mathbf{W}_{opt} &= (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\mathbf{Y} \\ &= \mathbf{X}^\dagger\mathbf{Y}, \end{aligned}$$

where $\mathbf{X}^\dagger$ represents the pseudo-inverse of $\mathbf{X}$. This is similar to the solution of the following equation:

$$\begin{aligned} \mathbf{Y} &= \mathbf{X}\mathbf{W} \\ &\Updownarrow \\ \mathbf{W} &= \mathbf{X}^{-1}\mathbf{Y}, \end{aligned}$$

which only holds if $\mathbf{X}$ is a non-singular square matrix. Because this assumption often does not hold, the pseudo-inverse is used instead. The approximated output can now be calculated as follows:

$$\hat{\mathbf{Y}} = \mathbf{X}\mathbf{W}_{opt}.$$

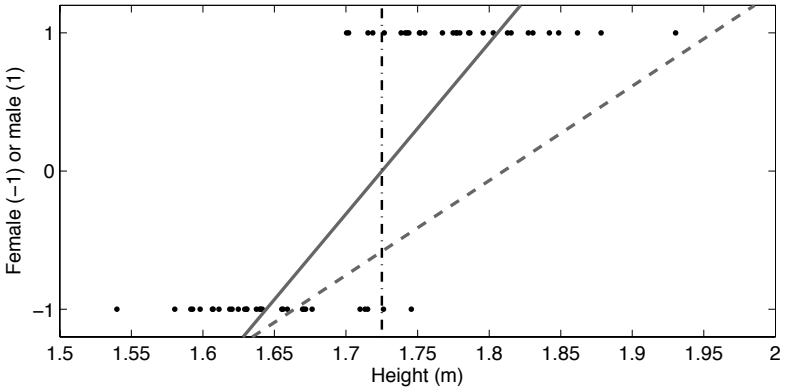


**Figure 2.2:** The height of humans as a function of their gender. The solid line represents the model trained using linear regression. The dashed line is the same model but for a dataset containing more women or a few very tall men. The dash-dot line is the optimal separation between the two classes.

## 2.3 Linear regression for classification

Although linear regression is designed to map a set of inputs to a line, it can also be used for classification. In the case of two classes one can map one class to $y = 1$ and the other class to $y = -1$. Figure 2.2 shows an example in which a person's gender is guessed from their height. For men the desired output value is 1, while it is $-1$ for women. Linear regression can now be used to draw a line through these points, the solid line in Figure 2.2, so that each height is now mapped to the estimated output $\hat{y}$. Although $\hat{y}$ is almost never $-1$ or 1, one can classify men and women with the simple rule that if $\hat{y} > 0$ the person is a man and $\hat{y} <= 0$ the person is a woman.

Because linear regression is not designed for classification it has several disadvantages. One of them occurs when one of the classes has more data points than the other. If there are for example more women than men in the training set, the line generated by linear

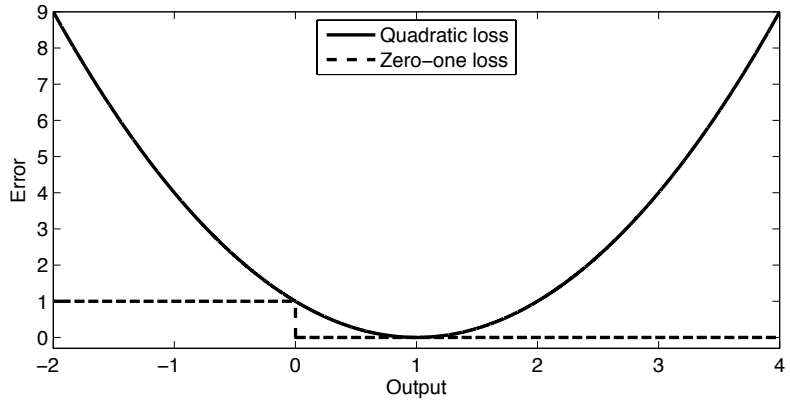


**Figure 2.3:** The quadratic loss and the zero-one loss for a desired output of 1.

regression will not be the solid line, but will be more like the dashed line in Figure 2.2. The same effect occurs if there are a few very tall men. Now the best separation between men and women is not found by determining whether $\hat{y} > 0$ or not. The best separation is found at $\hat{y} > \delta$ with in this case a threshold $\delta < 0$. It is obvious that $\delta$ is a parameter that needs to be optimized. Although this solution is only correct if there is only one input feature, it is an often used approximation for multiple input features. Changing the threshold $\delta$ is equivalent to changing the bias $w_0$.

These problems occur because the loss function that linear regression minimizes, is the quadratic error. Figure 2.3 shows the error function in a desired output of 1 (solid line). If a threshold of 0 is used you see that samples for which the output is larger than 0 and not equal to 1 are incorrectly punished. For classification however it is usually more opportune to give each correctly classified sample 0-error and each incorrectly classified sample an error greater than 0, for example 1. This error measure is called the zero-one loss and is plotted as a dashed line in Figure 2.3. To minimize this error, many ML techniques have been developed (Menard, 2002; Hsu et al., 2003). The memory use and training time of these techniques, however, has the disadvantage of scaling at least linearly with the size of the dataset.

Although detecting epileptic seizures is a classification task, it is

in practice not as simple as classifying each sample as correctly as possible. The error measures that are important for seizure detection are the number of false detections, the number of missed seizures and the detection delay of the detected seizures. A low number of correctly classified samples is in this case only an indication of good performance, but apart from that rather unimportant. Since the epilepsy datasets are often very large and there is no technique known in literature that optimizes the error measures relevant for seizure detection, several forms of linear regression are used. These are discussed in more detail in Chapter 3.

## 2.4 Non-linear regression

Although linear techniques are often powerful, there are many tasks for which they are not suited. These non-linear tasks require that the input features are processed in a non-linear manner. An example from seizure detection with two input features is illustrated in Figure 2.4. In this example the energy is measured in windows of 1 s on the onset EEG channel in the frequency ranges of 0 to 16 Hz and 16 to 40 Hz. It is clear that there are many misclassifications using linear regression.

An often used technique to perform better in non-linear tasks, is to apply some form of non-linear transformation $\phi$ of the input features. In this higher dimensional feature space, previously discussed linear techniques can be used. One of the simplest techniques to achieve this is called polynomial expansion or general linear model. If there are for example two input features $x_1$ and $x_2$ this is achieved by the following transformation:

$$\phi(x_1, x_2) = (x_1, x_2, x_1x_2, x_1^2, x_2^2, ..., x_1^2x_2^{n-2}, x_1x_2^{n-1}, x_1^n, x_2^n),$$

with $n$ the order of the transformation. For $n = 2$ this is reduced to:

$$\phi(x_1, x_2) = (x_1, x_2, x_1x_2, x_1^2, x_2^2).$$

Similar to what was discussed in the previous section, the output can

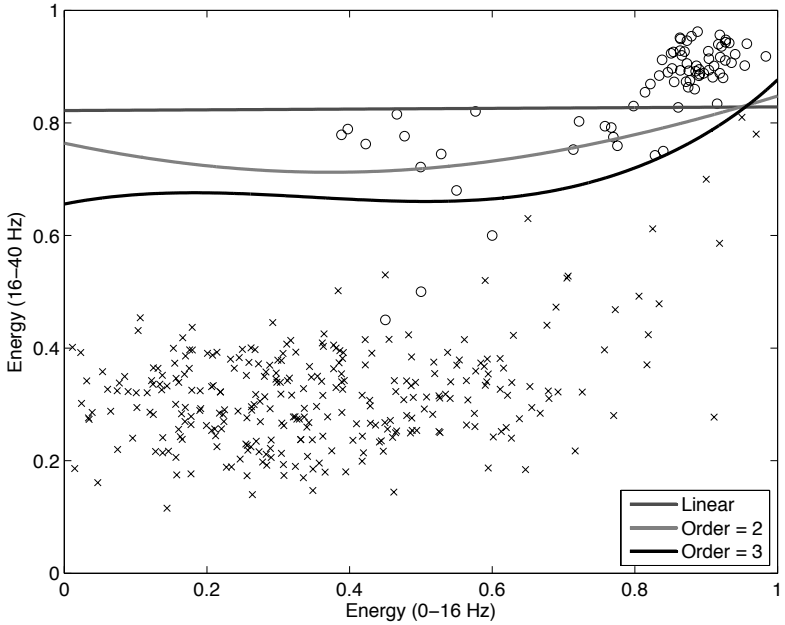


**Figure 2.4:** An example of a polynomial expansion to separate an epileptic seizure from normal EEG. The threshold is optimized to have no false positives.

now be generated with the following function:

$$f(x_1, x_2) = w_0 + w_1 x_1 + w_2 x_2 + w_3 x_1 x_2 + w_4 x_1^2 + w_5 x_2^2,$$

where the $w_i$'s are the weights of the linear function and $w_0$ represents the bias. As this simple example illustrates, the data that contains 2 features is now mapped to 5 features. This moves the data from the 2 dimensional input space to the higher, 5-dimensional feature space. For $n = 3$ this becomes a 9-dimensional feature space, for $n = 4$ this space is 14-dimensional, and so on.

In Figure 2.4 the decission boundaries for the second and third order polynomial expansion are shown. It is clear that with increasing system complexity the performance becomes better and better. This example actually illustrates what was shown in Cover (1965): mapping the input to a higher dimensional space increases the probability that the different classes can be linearly separated. However, making

a good prior assumption of the non-linearity in the model will result in better performance. If there is a polynomial relation between the in- and output, it is for example not the best possible solution to use an exponential model.

## 2.5 Over-fitting

When a student is asked to study for an exercise exam she has several options. Let us consider three students, each with their own learning strategy. The first student tries to understand the theory and exercises, and learns the reasoning behind them. The second student is not interested in the reasoning and learns the exercises by heart. The third student is a lazy student and learns only some of the reasoning. If the exam contains the exact same exercises as the ones seen in class, the first student will probably score slightly less than the second student. If, however, the exam contains different exercises the first student will have similar grades but the second student will fail miserably. For both exams, the third student will have his usual below average grades. What is called extreme over-fitting in ML can be compared with the learning strategy of the second student. The learning strategy of the third student can be seen as under-fitting. What you want in machine learning, however, is the strategy of the first student.

In most ML tasks the data is very noisy. Learning the noise as opposed to the task can be seen as over-fitting. Lets say you want an ML technique to learn the sine function shown in Figure 2.5, from a few noisy measurements. When the complexity of the polynomial expansion is increased to the 10-th order, you see that the the system was able to learn the data perfectly. If you compare this result to the actual sine wave it tries to fit, it is clear that lower order and less complex systems perform significantly better. The 10-th order model has in fact learned the noise on the input data.

To avoid over-fitting many strategies have been studied. In Chapter 3 the techniques used in this work are discussed. Each of these strategies makes a different prior assumption about the model. It can

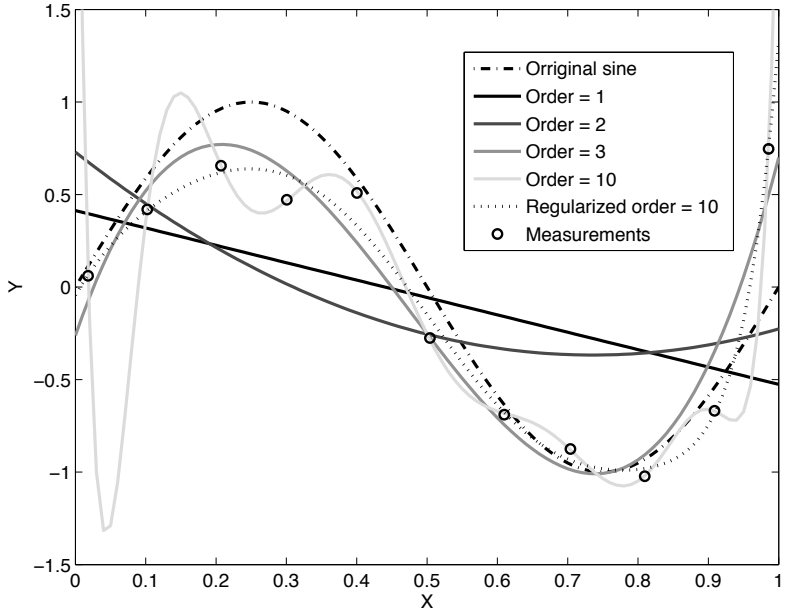


**Figure 2.5:** An example of a polynomial expansion to learn a sine wave.

be shown that, for linear regression, the size of the weights is related to the system complexity under the assumption that the input data is distorted by Gaussian noise. The technique most frequently used is called Tikhonov regularization (Tikhonov and Arsenin, 1977). It uses a single parameter, the regularization parameter, that scales the cost of the size of the weights, in order to keep them small. If, the most commonly used quadratic cost is applied, it is referred to as ridge regression (RR). For RR equation 2.1 becomes:

$$f_{loss}(x, y) = (y - w_1 x - w_0)^2 + \frac{\lambda}{M}(w_1^2 + w_0^2),$$

with $\lambda$ the regularization parameter and $M$ the number of training data points (which is only added for mathematical convenience). A more mathematically detailed explanation will be covered in Section 3.1.

With a lower regularization parameter, it is more probable that

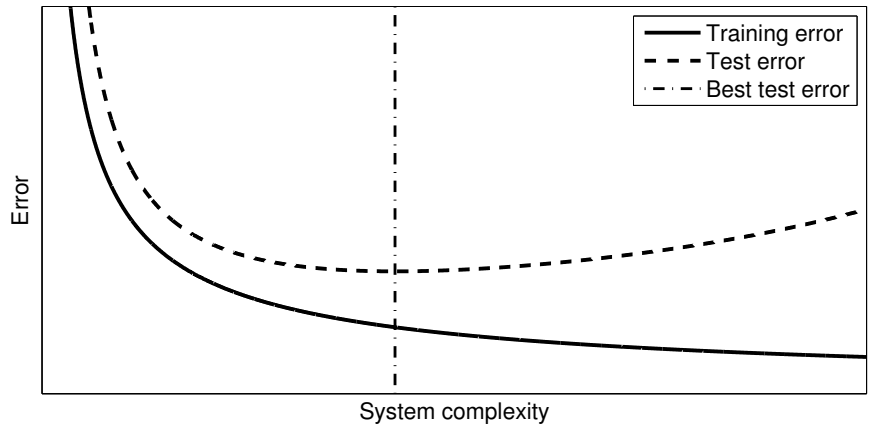


**Figure 2.6:** The train and test error as a function of the system complexity.

the system is over-fitting. The higher this parameter, the higher is the probability the system is under-fitting. The optimal regularization parameter can thus be found on the edge between over- and under-fitting. In the example shown in Figure 2.5, the dotted line represents a regularized version of the $10th$ order polynomial expansion. Note that this result is relatively close to what a human would draw with some prior knowledge of the smoothness of the curve, but without knowing that the function is a sine wave.

In Figure 2.6 one can see an example of the effect of over- and under-fitting on the performance. It shows how the training and test error evolve as a function of the system complexity. In ML, we want to try to minimize the test error as much as possible by training the system on a training set. If the system complexity is too low, both the train and test error are high. In this case the system is unable to find a good fit for the data. If you increase the system complexity you see that at some point the test error stops decreasing. This is the point where you reach the optimal fit with respect to the system complexity. After this point the system starts over-fitting and the test error increases even though the train error keeps decreasing.

Because RR assumes that the input data is distorted by Gaussian noise, it is obvious that it will not work as well as it should if

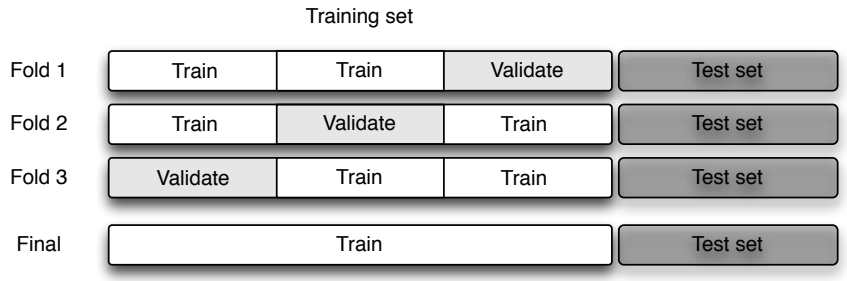


**Figure 2.7:** A schematic representation of 3-fold cross-validation.

the noise distortion is for example salt and pepper noise (Wang and Zhang, 1999). Regularization can also be achieved by making good model assumptions. If there is a polynomial relationship between the input and the output, it will be much easier to learn the input-output relation using a polynomial model than using an exponential model. Making a good prior assumption is thus paramount for the performance and the different assumptions made in this work will be discussed in Chapter 3.

## 2.6 Parameter optimization

ML techniques are often characterized by many 'hyper-parameters' such as the regularization parameter. These are parameters which the ML algorithm can not learn by itself and have to be determined in a different way. Changing these hyper-parameters often has a dramatic effect on the performance. Therefore it is necessary to optimize these parameters. At first glance one might come up with two options. Choose the parameters that give the best result on the training data. This can cause that these parameters only work well on the training data and you have caused the system to over-fit. What you want however is that the test results are optimal. But choosing the parameters that give the best test performance will not indicate how good the system performs on unseen data.

As a solution for these problems you can divide the training set in two parts. One part for training and one part to validate the performance of the parameters. If this is repeated several times for different train and validation sets, it is called cross-validation. Many cross-validation techniques exist (Kohavi et al., 1995), the most common used is n-fold cross-validation. In Figure 2.7 an example is shown of 3-fold cross-validation, where the train set is subdivided in 3 parts. For each fold the system is trained on 2 of the 3 parts for all the possible parameters and the error is validated on the third part. This is repeated for each fold and finally the parameters that resulted in the minimal sum over the errors are selected. This technique is often used in ML and most commonly 10-fold cross-validation is applied. After the best parameters are found, the full training set can be used to finally train the system. That way all the available data is used together with a near optimal set of parameters.

There is one condition for this optimization technique, and for most ML techniques for that matter, to work. The data in the training and test sets must be very similar and thus samples from the same data distribution. If you train for example a system to classify images of reptiles and amphibians, and if the training set only contains images of snakes and frogs, the system will probably not be able to classify crocodiles and salamanders. Techniques from the field of semi-supervised learning, TL and/or AL have been shown to be able to surpass this problem to a certain extent (Pan and Yang, 2010; Krogh and Vedelsby, 1995).

## 2.7 Artificial neural networks

Artificial neural networks (ANNs) are an ML technique which is based on an extremely simplified model of the brain. The artificial neurons are represented by a very basic mathematical model. Most commonly, as shown in Figure 2.8, a weighted sum of the inputs followed by a non-linear function, usually a hyperbolic tangent (see Figure 2.11), is used. These neurons are then stacked in 2 or more layers as shown in Figure 2.9: one but possibly more hidden layers and an output layer.

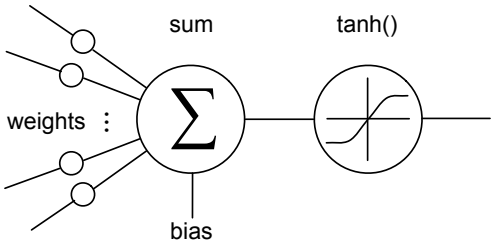


**Figure 2.8:** A schematic of the artificial neuron most commonly used in an artificial neural networks

The input is propagated through the hidden layer(s) to the output. To train these networks, the weights are first randomly initialised. Next the data is fed through the network and an error gradient is calculated between the generated and desired output. Using this error gradient, the weights are adapted. This process is repeated until the system converges, or until the maximum number of steps is reached. In literature it is known as back-propagation and for more details we refer to Chauvin and Rumelhart (1995).

These feed-forward artificial neural networks are known for their generalization properties and ability to learn complex relationships between inputs and outputs with limited training. When recurrent connections are added, shown as dashed lines in Figure 2.9, the model gains information from the past. Previous inputs can remain present in the dynamics of this network and will influence the current output. This results in a slightly more biologically relevant dynamical system that can be taught to find relationships between the desired output and any past input. However, he number of neurons $N$ limits the total memory that can be available in the network. Inputs from more than $N$ time-steps in the past can only be partially remembered. Traditionally, all interconnection weights between the neurons in these recurrent artificial neural networks are trained using back-propagation through time. It unfolds the network in time so that back-propagation, as it is used in feed-forward artificial neural networks, can be applied. However, this technique is characterised by often long training times and stability issues. For more information

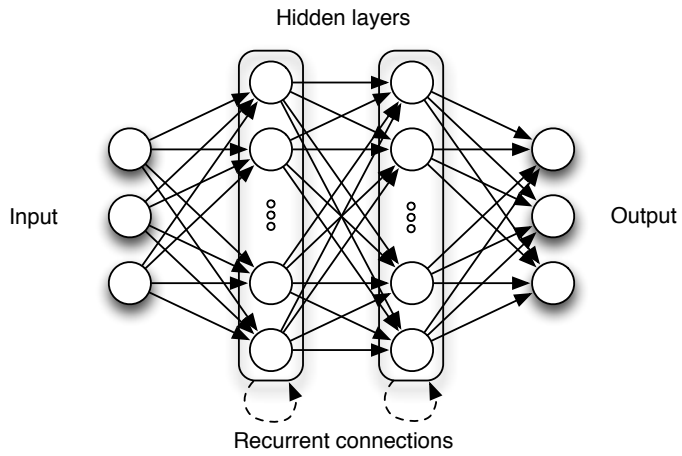


**Figure 2.9:** A schematic of a feed-forward and recurrent neural network. The dashed lines are recurrent connections which are not used in feed-forward neural networks.

on this complex but very powerful training technique we refer again to literature (Pineda, 1987).

## 2.8 Reservoir computing

Reservoir computing (RC) (Verstraeten et al., 2007) is a training technique for recurrent artificial neural networks and a generalization of the echo state network approach introduced in (Jaeger, 2001). As opposed to recurrent artificial neural networks where all the weights are trained using back-propagation through time, RC uses a randomly created recurrent network, called a reservoir which is left untrained. From this network, which is illustrated in Figure 2.10, only a linear output is trained. That way the long training time and stability issues of regular recurrent artificial neural networks are avoided without losing the desired generalization abilities (Jaeger, 2002).

In the RC set-up each non-zero input sample will excite this dynamical system and push the reservoir to a new state. In practice this can be seen as a projection of the input features to a higher dimen-

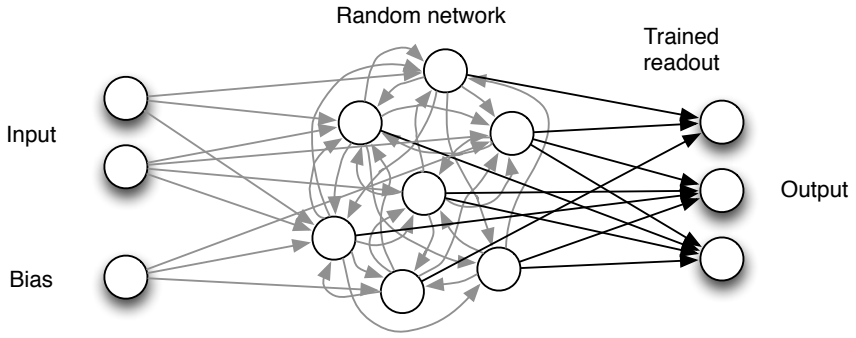


**Figure 2.10:** A schematic representation of reservoir computing. The gray arrows represent the untrained and randomly created connections. The black arrows represent the trained linear readout.

sional feature space comparable to non-linear regression. The biggest difference here is that inherently in the system there is information from the past inputs which is slowly forgotten with an exponential decay, as will be illustrated in Figure 2.15 (Hermans and Schrauwen, 2010).

In the next section it is explained in more mathematical detail how RC works. In Section 2.8.2 a more comprehensible approach is taken to show how the parameters influence the performance of RC and how they should be changed in order to achieve the best possible results. For even more details we refer the reader to Verstraeten (2009). Finally, in Section 2.8.3, a link is made between RC and other ML techniques.

## 2.8.1 Mathematical description

The operation of the reservoir is shown in Figure 2.10 and can be described as follows. We use $\mathbf{x}[k]$ to represent the current activation values of the neurons in the reservoir at time $k$, $\mathbf{u}[k]$ as the input vector, $\mathbf{y}[k]$ for the desired output and $\hat{\mathbf{y}}[k]$ for the approximated output generated by the RC system. The inputs of the neurons in the reservoir are connected with the input, a constant input called the

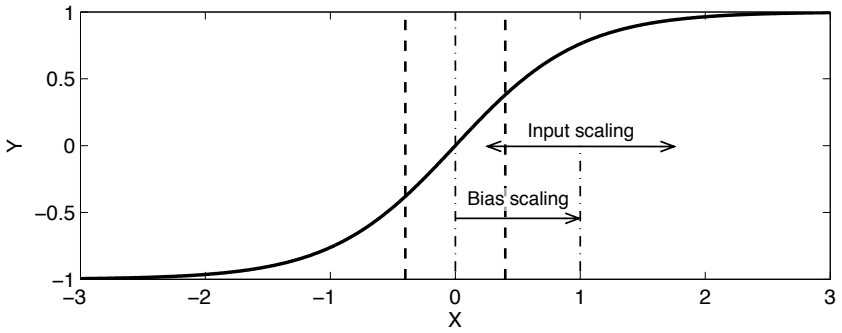


**Figure 2.11:** The hyperbolic tangent function together with a schematic representation of the influence of the input and bias scaling. Between the dashed lines can be considered as the linear operation area, whereas outside this area the function can be considered as non-linear.

bias and the output of all the neurons in the reservoir as illustrated in Figure 2.10. The weights of these connections are represented by the weight matrices $\mathbf{w}_{bias}$ $(N \times 1)$, $\mathbf{W}_{inp}$ $(N \times n)$ and $\mathbf{W}_{res}$ $(N \times N)$, with the dimensions between brackets, $N$ the number of neurons and $n$ the number of inputs, respectively.

Typical for RC is that these matrices are randomly initialised and are not changed during training. Most commonly, the following initialization is performed:

- The elements of the bias weight matrix $\mathbf{w}_{bias}$ are uniformly distributed between $-1$ and $+1$.
- The internal weight matrix $\mathbf{W}_{res}$ is initialised using a Gaussian distribution with a mean of 0 and a standard deviation of 1.
- All elements in the input weight matrix $\mathbf{W}_{inp}$ are randomly set to $-1$ or $+1$.

The sparseness of these matrices as well as the initialization process are, however, not critical for the performance.

If basic sigmoid neurons are used, a weighted sum followed by the hyperbolic tangent function (shown in Figure 2.11), the state update

equation is given by:

$$\mathbf{x}[k+1] = \tanh(\mathbf{W}_{res}\mathbf{x}[k] + \mathbf{W}_{inp}\mathbf{u}[k] + \mathbf{w}_{bias}).$$

Although most commonly the hyperbolic tangent function is used as a non-linear function, any function or neuron type can be used. In this work leaky integrator neurons are used, i.e. basic sigmoid neurons followed by a first-order low-pass filter. The state equation now becomes:

$$\mathbf{x}[k+1] = (1-\gamma)\mathbf{x}[k] \ + \gamma \tanh(\mathbf{W}_{res}\mathbf{x}[k] + \mathbf{W}_{inp}\mathbf{u}[k] + \mathbf{w}_{bias}).$$

In this equation $\gamma$, the leak rate, represents the rate at which the previous reservoir state is 'leaked' and replaced by the new reservoir state. It sets the cutoff frequency of the low-pass filter in the neurons. This extra parameter of leaky integrator neurons is used to tune the reservoir memory and timescales (Jaeger et al., 2007).

The previous equations show how the input features are mapped to a higher dimensional feature space. To generate the output from these reservoir features, a form of linear regression is usually applied:

$$\hat{\mathbf{y}}[k] = \mathbf{W}_{out} \begin{bmatrix} \mathbf{x}[k] \\ 1 \end{bmatrix},$$

in which '1' represents the output bias. The weights of the output weight matrix $\mathbf{W}_{out}$ are trained using a form of linear regression such as RR.

### 2.8.2 Parameters

Any dynamical system can be considered as a reservoir (Verstraeten et al., 2010). In the formulation used above, there are 4 parameters that set the dynamics of the reservoir, each with their own influence (Jaeger, 2002). Since the connection weights in the reservoir are randomly generated, 3 of the parameters scale these weights. The fourth parameter is the leak rate which was briefly mentioned in the previous section. Because every task has its own specific needs, these parameters need to be optimized for each task independently and can

have great influence on the performance of RC. When using RC it is therefore of great importance to understand and optimize these parameters. Apart from the 4 parameters that determine the dynamics, the reservoir size is a fifth parameter that needs to be determined, which is, however, not optimized.

## Spectral radius

The spectral radius is a scaling factor for the connection weights between the neurons. It represents the largest absolute eigenvalue of the connection weight matrix $\mathbf{W}_{res}$. As shown in Figure 2.12, this parameter has a significant influence on the dynamics. Linear reservoirs with a spectral radius larger than or equal to 1 can be unstable since the reservoir states do not die out over time as shown in Figure 2.12. If the spectral radius is smaller than 1, a linear reservoir is said to have the echo state property (Jaeger, 2002), which means that the input fades away like an echo. In non-linear reservoirs this rule is not always true. The most commonly used hyperbolic tangent function for example, (shown in Figure 2.11) limits the reservoir states once they reach the non-linear area. In extremis, they will saturate to -1 and 1. This implies that the spectral radius influences the non-linearity of the reservoir dynamics. Typical values for optimizing the spectral radius range from 0.4 to 1.4 in steps of 0.1.

## Input scaling

The input scaling parameter can be used to set the influence of the input to the reservoir states as shown in Figure 2.12. The larger the input scaling, the bigger the effect of the input on the reservoir states. With a smaller input scaling the current reservoir dynamics will be less disturbed by a new input. The input scaling also has an influence on the non-linearity because of the hyperbolic tangent function. As illustrated in Figure 2.11, it determines the area of the hyperbolic tangent function that is covered by the input. The input scaling is highly dependent on the amplitude of the input, and should be chosen on a logarithmic scale. Typically for normalized inputs it ranges from 0.001 to 0.1 and is optimized in steps of $10^{0.2}$ on a logarithmic scale.

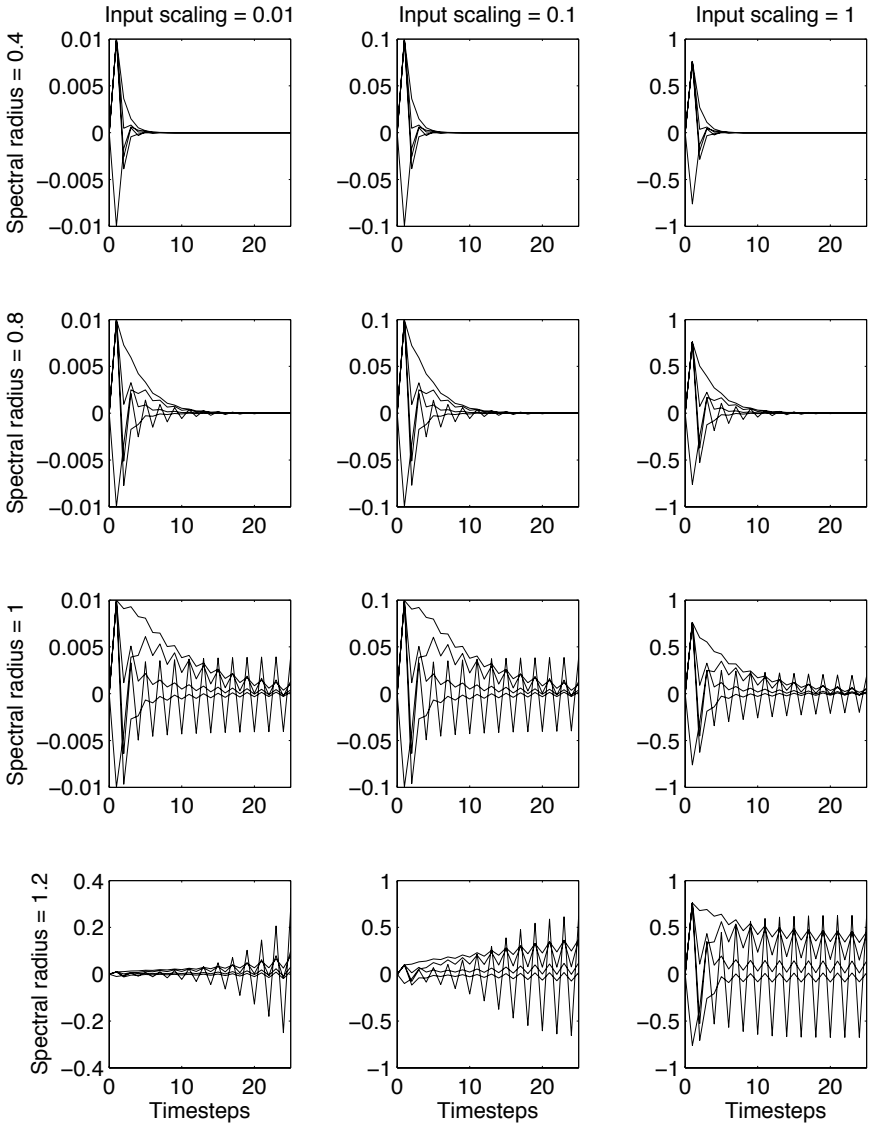


**Figure 2.12:** The evolution of the reservoir state as a function of the spectral radius and the input scaling. The reservoir contains 5 neurons and as input a pulse with amplitude 1 and a length of 1 time-step is used. The bias scaling and the leak rate have been set to 0 and 1, respectively.

## Bias scaling

The bias scaling pushes the reservoir states closer to 1 or -1 as illustrated in Figure 2.13. This corresponds to the non-linear region of the hyperbolic tangent shown in Figure 2.11. Hence, with a higher bias scaling the reservoir becomes more non-linear. The non-linear region of the hyperbolic tangent has a smoothing effect, such that reservoirs with a spectral radius greater than 1 can have fading dynamics as long as the bias is high enough. Typical values for the bias are 0, 0.1 and 1. Theoretically it should be optimized on a logarithmic scale, but in practice, using different values than the previously mentioned has nearly no effect on the performance.

## Leak rate

The leak rate is usually implemented as an extra recurrent connection in the reservoir that is essentially a low-pass filter on the reservoir states (Verstraeten et al., 2007). It represents the rate at which the previous reservoir state is replaced by the current reservoir state, such that a leak rate of 1 represents a reservoir without the low-pass filter. As shown in Figure 2.14, it smooths the reservoir states over time. Just like the bias it has an effect on the fading of the reservoir dynamics and can push reservoirs with a spectral radius larger than 1 into a stable regime. From Figure 2.14 one might conclude that the leak rate has more influence on the stability than the bias, but this is not the case. The input used in Figure 2.14 is a simple step function which is zero most of the time. It generates an oscillation with a frequency above the cut-off frequency of the low pass filter implemented by the lowest leak rates. An input signal with a frequency component significantly below the cut-off frequency will not be dampened in a linear reservoir. The influence of the leak rate is therefore frequency dependent. Typically this parameter is logarithmically optimized in steps of $10^{0.25}$. Since this parameter is very dependent on the time scale of the task, there is no typical range within which the leak rate should be optimized.

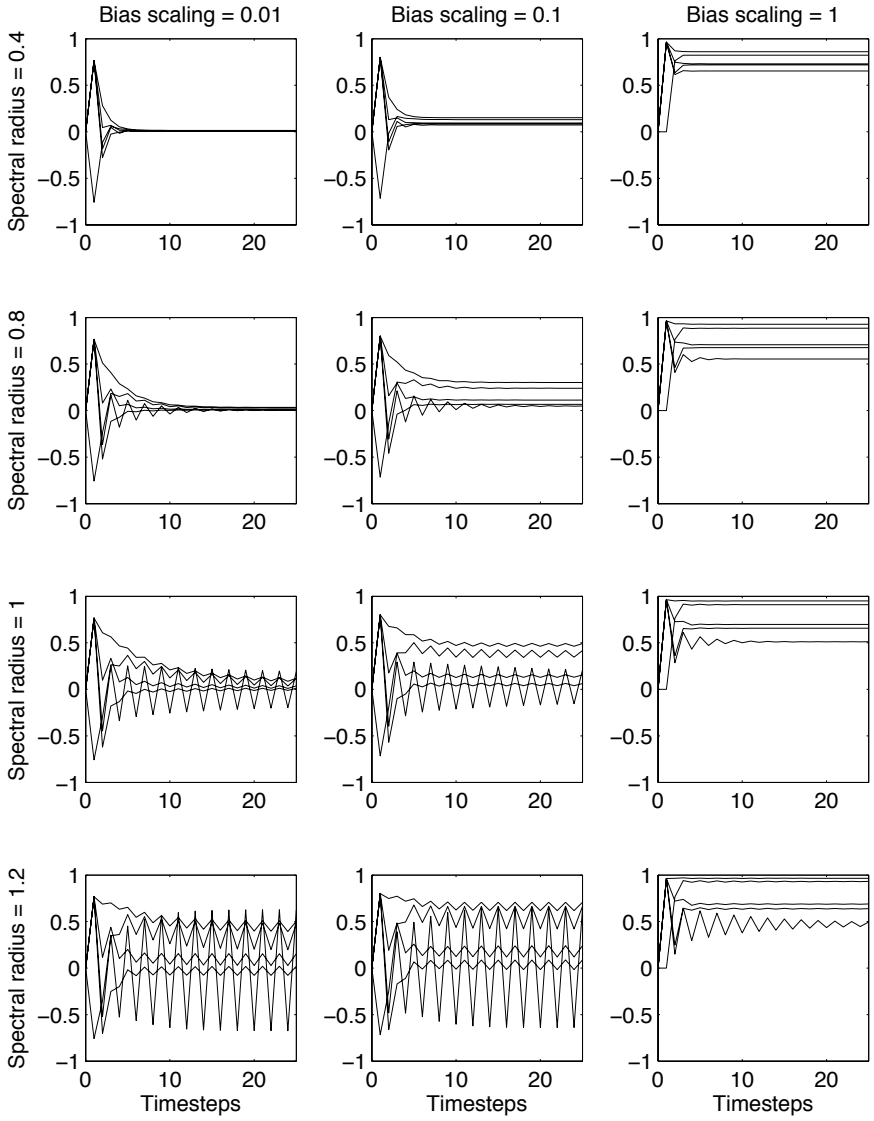


**Figure 2.13:** The evolution of the reservoir state as a function of the spectral radius and the bias scaling. The reservoir contains 5 neurons and as input a pulse with amplitude 1 and a length of 1 time-step is used. The input scaling and the leak rate have both been set to 1.

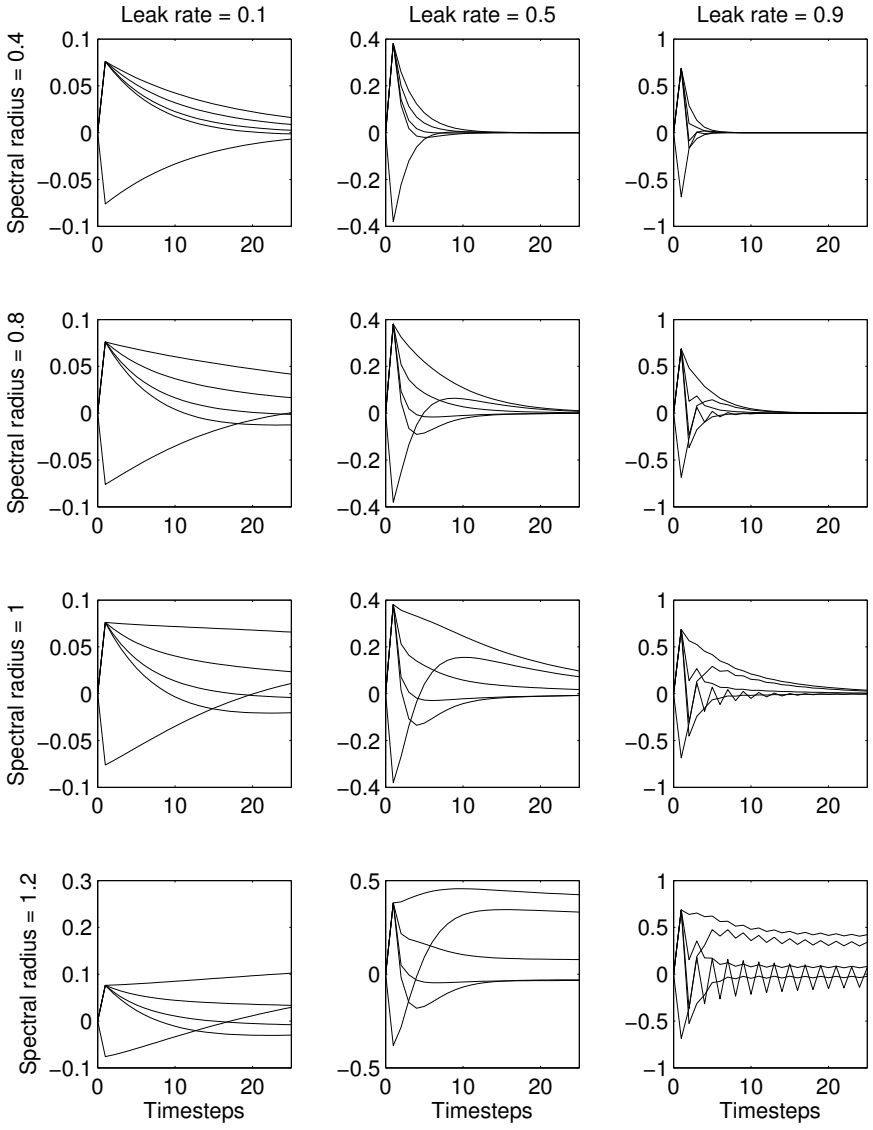


**Figure 2.14:** The evolution of the reservoir state as a function of the spectral radius and the leak rate. The reservoir contains 5 neurons and as input a pulse with amplitude 1 and a length of 1 time-step is used. The input scaling and the bias scaling have been set to 1 and 0, respectively.

## Reservoir size

The reservoir size has a lot of impact on the performance since it is related to the model complexity and the system memory. As a general rule one could say that with a larger reservoir, the performance increases if and only if the system is properly regularized. However, this performance gain will decrease significantly with larger and larger reservoir sizes. When the system is not regularized, over-fitting will occur from a certain reservoir size on. Increasing the reservoir size has one obvious disadvantage since the computational cost increases. If leaky integrator neurons are used, it scales quadratically with the reservoir size. Therefore the reservoir size is usually chosen at a point where increasing the size has little to no effect on the performance.

This correlation between the reservoir size and the performance has two causes. First and most importantly, as shown in Section 2.4, the performance increases when the input is mapped to a higher dimensional space (Cover, 1965). Secondly with a larger reservoir the memory of the system increases as shown in Figure 2.15 (Hermans and Schrauwen, 2010). One way to quantify this, is by measuring the reservoir's linear memory capacity. This measure expresses how well a system can reconstruct past input values from its state, in other words, how well can you train a reservoir to produce delayed versions of its input signal. The memory capacity is however strongly associated to the non-linearity (Verstraeten et al., 2010). The more non-linear the reservoir, the shorter the memory. Tasks that require a long memory either require a very linear reservoir or a very large reservoir. As mentioned before, the spectral radius defines how long it lasts before the past inputs fade away. Therefore, reservoirs with a very small spectral radius have a shorter memory as shown in Figure 2.15.

## Optimizing the parameters

Although each parameter has its own specific effect on the reservoir, Figures 2.12 to 2.14 also show that multiple parameters can have an effect on one reservoir property. The processing power of reservoirs for example, is shown to be largest when operating at the edge of stability (Legenstein and Maass, 2007) and although the spectral radius

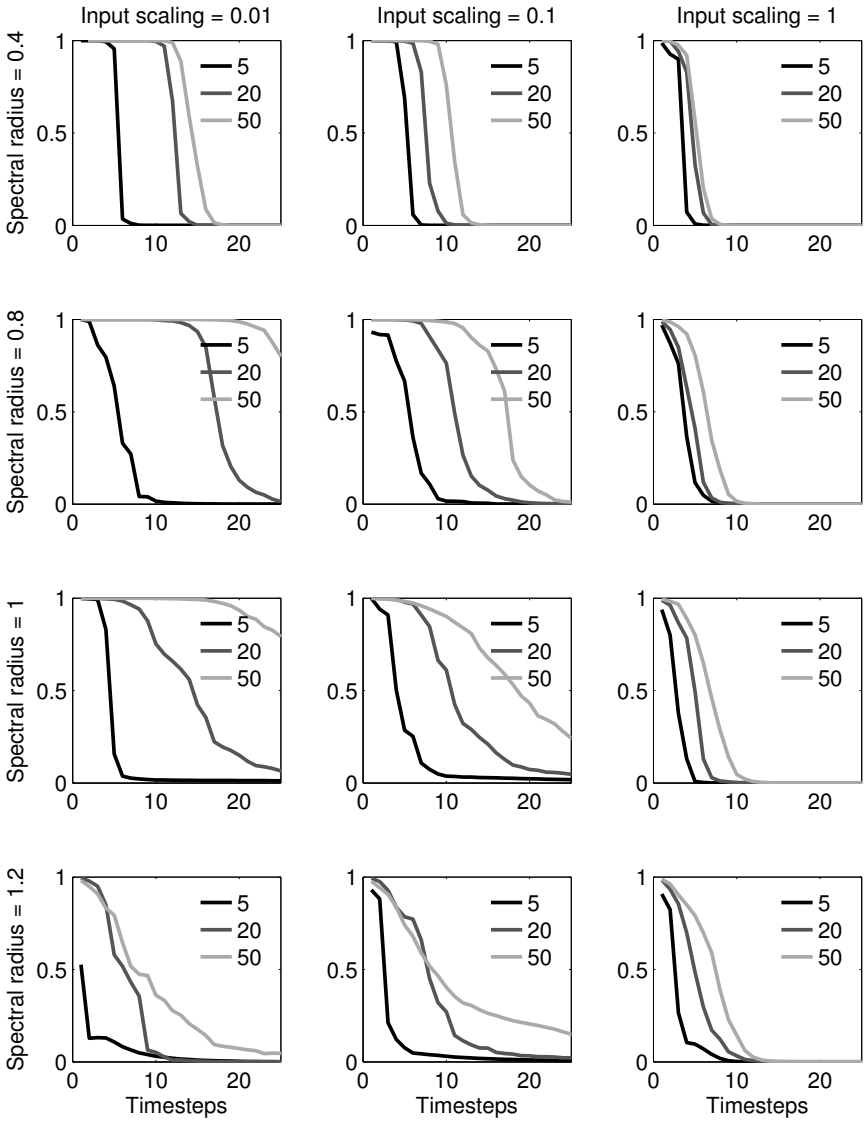


**Figure 2.15:** The linear memory capacity of reservoirs with 5, 20 and 50 neurons versus the number of time-steps in the past, as a function of the input scaling and the spectral radius. The area under the curves represents the total memory capacity. The bias scaling and the leak rate have been set to 0 and 1, respectively.

has the most influence, it also depends on the other parameters. To select the optimal set of parameters cross-validation is usually applied. Because the reservoir weights are randomly generated, this process is repeated for several reservoir initialisations, usually around 10 different initializations are used. Optimally the 4 parameters that influence the reservoir dynamics, are optimized in one big grid search. This is however very computationally expensive, so usually the parameters are optimized one by one. In this work only the spectral radius and the input scaling are optimized simultaneously. Because of the interdependence between the parameters, this process is repeated several times starting with a very rough grid of parameters and moving towards a smaller and smaller step size.

### 2.8.3 Link with other machine learning techniques

Many ML techniques train a direct mapping from the input to the output, similar to linear or non-linear regression. In these techniques, the current output is independent of the previous one. In order to give this type of system information about the past, a time window is often used. Using this technique, not only the current input is used but also the $N$ past inputs, where $N$ is the depth of the time window. This model however assumes that only the past $N$ inputs have an influence on the output.

In RC a model is built for which past inputs have less influence on the current output, if they occurred longer ago (Lukoševičius and Jaeger, 2009; Verstraeten et al., 2010). It has a forgetting factor built in with a quasi exponential decay (see Figure 2.15). This fading memory prior turns out to be a very effective assumption for many tasks. In seizure detection for example a rhythmic burst is only relevant if it has been recently preceded by spikes or other activity that is specific for a seizure onset. However if these spikes have occurred many minutes in the past, they might be inter-ictal spikes and irrelevant for seizure detection.

Similarly to other ML techniques such as kernel based methods, RC makes a non-linear mapping of the input to a higher dimensional

space (Lukoševičius and Jaeger, 2009). It can be compared to time window based non-linear techniques with a fading memory prior, such that past inputs have less influence on the output if they occurred longer ago.

Support vector machines (SVMs) are a very commonly used technique in ML (Hsu et al., 2003). Here a so called kernel is used which maps the input features to a virtually infinite dimensional space where their inner product is computed. Thanks to a mathematical trick, called the kernel trick, the inner product can be computed without actually mapping the data to the infinite dimensional space. It is therefore not trained in this feature space, but in the so called dual-space. During training, the most relevant data points from the training set, the support vectors, are selected. After training every new data point is 'compared' with these support vectors using a form of linear regression which generates the output. One can show that it is possible to create a recurrent kernel that is equivalent to an infinite sized reservoir (Hermans and Schrauwen, 2012).

Since in RC a random recurrent neural network is used, it is not illogical to apply the same strategy to forward neural networks. Extreme learning machines (ELMs) are in fact the forward equivalent of RC, where a random hidden layer is created. After this hidden layer, linear regression is used as opposed to back-propagation to train the output. Although RR can be easily applied to avoid over-fitting, an other technique is often applied. In optimally pruned ELMs (OP-ELMs), the model complexity is reduced by removing neurons from the hidden layer. A similar technique can be applied to RC (Dutoit et al., 2009) for which in the next chapter a mathematically efficient algorithm is derived.

RC is however not fully equivalent to a non-recursive SVM or ELM combined with a fading memory prior. The recursive connections in the reservoir not only make sure that the past is forgotten, but also that the past inputs are processed more non-linearly for each iteration. Apart from the computationally expensive recursive SVMs presented in (Hermans and Schrauwen, 2012), there is no one to one mapping to other ML techniques, which makes that RC deserves its own rightful place in the ever growing list of ML techniques.

Any (non-linear) function or even neuron type can be used in the

reservoir (Verstraeten et al., 2007; Lukoševičius and Jaeger, 2009). One could for instance use single atoms and apply their non-linear reactions with each other as computing power. This means that even a bucket of water can be used as a reservoir for speech recognition (Fernando and Sojakka, 2003). Or that lasers and optical wave guides can be used to compute (Vandoorne et al., 2008).

## 2.9 Conclusion

This chapter covered a short introduction to ML and the most common learning strategies such as linear regression. Terms like active learning, non-linear transformations, over-fitting and cross-validation were explained. Next a broad introduction to RC was given: how it works and how the get the best performance. Finally the functionality of RC was compared with other ML techniques.

# 3
# Optimized regularization techniques

As mentioned in Section 2.5 regularization is needed to avoid overfitting. In this chapter, the techniques used in this work will be discussed in more detail. Each of these techniques makes a prior assumption over the model and the data. Using the correct assumption will result in a model that is more properly regularized and will perform better on the task. Since the EEG datasets used in this work are so large, an algorithm is given for each of the regularization techniques that is optimized for large datasets. Sections 3.1 to 3.3 cover newly developed optimization algorithms for existing regularization techniques (Buteneers et al., 2012a). Section 3.4 introduces a new regularization technique (Buteneers et al., 2012b). Implementations of these algorithms can be found in the Oger toolbox (Verstraeten et al., 2011).

This is a more technical chapter that covers the adaptations that were done to the default RC training technique. The introductions or the first paragraphs of each section have been written to give readers a grasp of the matter discussed in this chapter. Subsections and mathematical equations can be skipped without missing the essential pieces of the puzzle to apprehend the following chapters. Each of the techniques has been introduced to provide a solution to the problems that occur in seizure detection: large datasets, regularization, class imbalance and different seizure signatures. In the next chapters these techniques will be applied and tested on seizure detection tasks.

# 3.1 Regularisation parameter optimization algorithm for ridge regression

As mentioned in Section 2.5, ridge regression (RR) assumes that the input data is distorted by Gaussian noise. Optimizing the regularization parameter however, can be a computational burden. Especially since this process needs to be redone for every random initialization of the reservoir. For small datasets many algorithms have been proposed to speed-up the optimization process (Cawley and Talbot, 2004; Pahikkala et al., 2006). However, they are not suited for large datasets since their computational complexity and memory use scale quadratically with the size of the dataset. In what follows a new algorithm is proposed based on an eigendecomposition that has been specifically designed to cope with the large datasets used in this work. It will be compared to two existing algorithms from literature: a naive implementation and an algorithm based on covariance matrices.

## 3.1.1 Naive implementation

In matrix notation RR minimizes the following loss function:

$$f_{loss} = ||\mathbf{X}_t\mathbf{w}_t - \mathbf{y}_t||^2 + \lambda||\mathbf{w}_t||^2, \tag{3.1}$$

where $\mathbf{X}_t$ represents the training input data, $\mathbf{y}_t$ the desired output on the train data, $\mathbf{w}_t$ the output weights and $\lambda$ the regularization parameter that adds an extra cost proportional to the $L_2$-norm of the output weights. For mathematical convenience and since each desired output requires a different regularization parameter, we assume only 1 desired output throughout the rest of this work. Minimizing Equation (3.1) results in the following equation for the output weights:

$$\mathbf{w}_{t,opt} = (\mathbf{X}_t^T\mathbf{X}_t + \lambda\mathbf{I})^{-1}\mathbf{X}_t^T\mathbf{y}_t. \tag{3.2}$$

If the optimal regularization parameter is known, calculating the optimal weights is of the order $O(N^2M + N^3)$ (Press, 1992), with $N$ the dimensionality of the input and $M$ the number of data samples in the train set. Here $O(N^2M)$ is the order to compute the covariance matrix $\mathbf{X}_t^T\mathbf{X}_t$ and $O(N^3)$ is the order to compute the matrix inverse. The computational cost of the other operations can be ignored with respect to these two.

To find the optimal regularization parameter $\lambda$, a list of possible $\lambda$s is created and through cross-validation the best $\lambda$ is selected. If the train set is representative for the test set, one can assume that when the validation error is minimized the test error gets reduced. The loss function on the validation set is defined as follows:

$$f_{loss,v}(\lambda) = ||\mathbf{X}_v\mathbf{w}_t(\lambda) - \mathbf{y}_v||^2, \tag{3.3}$$

where $\mathbf{X}_v$ represents the validation data and $\mathbf{y}_v$ the desired output on the validation data. Typically, an output weight matrix is trained for each train set and each regularization parameter and then the validation error is calculated using Equation (3.3). If $R$ represents the number of regularization parameters and $K$ the number of validation sets, the computational complexity of this procedure becomes of the order $O(RK(N^2M + N^3))$. The datasets used in this work are typically very large, such that $N \ll M$. This means that $N^2M \gg N^3$ and thus the cost of the matrix inversion can be ignored. Thus for large datasets the computational complexity becomes $O(RKN^2M)$.

### 3.1.2 Covariance method

To speed up the optimization of the regularization parameter we can rewrite Equation (3.3) as follows:

$$\begin{aligned} f_{loss,v,\lambda} &= \mathbf{w}_t^T\mathbf{X}_v^T\mathbf{X}_v\mathbf{w}_t - 2\mathbf{w}_t^T\mathbf{X}_v^T\mathbf{y}_v + \mathbf{y}_v^T\mathbf{y}_v \\ &= \mathbf{w}_t^T(\mathbf{X}_v^T\mathbf{X}_v\mathbf{w}_t - 2\mathbf{X}_v^T\mathbf{y}_v) + \mathbf{y}_v^T\mathbf{y}_v \\ &= \mathbf{w}_t^T(\mathbf{A}_v\mathbf{w}_t - 2\mathbf{b}_v) + c_v, \end{aligned} \tag{3.4}$$

where $\mathbf{A}_v$ denotes the covariance matrix of the validation data, $\mathbf{b}_v$ the cross-covariance vector of the validation data and the desired out-

put and $c_v$ the variance of the desired output. Since each output is considered independently, these covariance matrices have a size of respectively $N \times N$, $N \times 1$ and $1 \times 1$.

Following the technique presented in De Brabanter et al. (2010) to optimize the regularization parameter for FS-LS-SVMs, the covariance matrix $\mathbf{A}$ on the train and validation set combined can be calculated as follows:

$$\begin{aligned}\mathbf{A} &= \mathbf{X}^T\mathbf{X} \\ &= \begin{pmatrix}\mathbf{X}_t \\ \mathbf{X}_v\end{pmatrix}^T \begin{pmatrix}\mathbf{X}_t \\ \mathbf{X}_v\end{pmatrix} \\ &= \mathbf{X}_t^T\mathbf{X}_t + \mathbf{X}_v^T\mathbf{X}_v \\ &= \mathbf{A}_t + \mathbf{A}_v.\end{aligned}$$

Analogously for $\mathbf{b}$, $\mathbf{w}_t$ can be rewritten as follows:

$$\begin{aligned}\mathbf{w}_t &= (\mathbf{A}_t + \lambda\mathbf{I})^{-1}\mathbf{b}_t \\ &= (\mathbf{A} - \mathbf{A}_v + \lambda\mathbf{I})^{-1}(\mathbf{b} - \mathbf{b}_v).\end{aligned} \tag{3.5}$$

The covariance matrices, which need to be calculated to find the optimal weights using Equation 3.2, can be calculated for each validation set before starting the validation procedure. This process has a computational cost of the order $O(N^2M)$. These covariance matrices need to be computed even when the regularization parameter does not need to be optimized so that it can be seen as a fixed computational cost to RR. To compute the $\mathbf{w}_t$-matrix, there is 1 matrix inversion needed for each $\lambda$, which is in standard implementation of the order $N^3$ (Press, 1992). Computing the covariance matrices and optimizing the regularization parameter now becomes of the order $O(N^2M + RKN^3)$, with R equal to the number of regularization parameters. This is significantly less than the cost for the naive implementation. Throughout the rest of this work we will refer to this technique as the covariance method.

### 3.1.3 Eigen method

In order to reduce the computational cost, a new approach is proposed. Determining $\mathbf{w}_t$ using the diagonalization or eigendecomposition of the real and symmetric covariance matrix $\mathbf{A}_t = \mathbf{A} - \mathbf{A}_v = \mathbf{C}_t\mathbf{D}_t\mathbf{C}_t^T$ (Parlett, 1980), gives:

$$\begin{aligned}\mathbf{w}_t &= (\mathbf{C}_t\mathbf{D}_t\mathbf{C}_t^T + \lambda\mathbf{I})^{-1}(\mathbf{b} - \mathbf{b}_v) \\ &= (\mathbf{C}_t\mathbf{D}_t\mathbf{C}_t^T + \mathbf{C}_t(\lambda\mathbf{I})\mathbf{C}_t^T)^{-1}(\mathbf{b} - \mathbf{b}_v) \\ &= \mathbf{C}_t(\mathbf{D}_t + \lambda\mathbf{I})^{-1}\mathbf{C}_t^T(\mathbf{b} - \mathbf{b}_v). \end{aligned} \tag{3.6}$$

Let us now compute the following matrices for each validation set beforehand:

$$\begin{aligned}\mathbf{b}_{Ct} &= \mathbf{C}_t^T\mathbf{b}_t = \mathbf{C}_t^T(\mathbf{b} - \mathbf{b}_v) \\ \mathbf{A}_{Cv} &= \mathbf{C}_t^T\mathbf{A}_v\mathbf{C}_t \\ \mathbf{b}_{Cv} &= \mathbf{C}_t^T\mathbf{b}_v.\end{aligned}$$

For each $\lambda$ the following can now be calculated:

$$\begin{aligned}\mathbf{w}_{Ct} &= (\mathbf{D}_t + \lambda\mathbf{I})^{-1}\mathbf{C}_t^T(\mathbf{b}_t) \\ &= (\mathbf{D}_t + \lambda\mathbf{I})^{-1}\mathbf{b}_{Ct}.\end{aligned} \tag{3.7}$$

Since both $\mathbf{D}$ and $\mathbf{I}$ are diagonal matrices and $\mathbf{b}_{Ct}$ is a vector, the previous calculation can be done element-wise and is thus of the order $N$. If we integrate this in Equation (3.4) we get:

$$\begin{aligned}f_{loss,v,\lambda} &= \mathbf{w}_{Ct}^T(\mathbf{C}_t^T\mathbf{A}_v\mathbf{C}_t\mathbf{w}_{Ct} - 2\mathbf{C}_t^T\mathbf{b}_v) + c_v \\ &= \mathbf{w}_{Ct}^T(\mathbf{A}_{Cv}\mathbf{w}_{Ct} - 2\mathbf{b}_{Cv}) + c_v.\end{aligned}$$

Since $\mathbf{A}_{Cv}$ is a square matrix of size $N \times N$ and the other matrices are vectors of size $N$, this is a vector-matrix multiplication of the order $O(N^2)$. Therefore this eliminates the matrix inversion of Equation (3.5) for each $\lambda$ which is of the order $O(N^3)$. For each validation set however one eigendecomposition of a symmetric matrix needs to be calculated which is of the order $O(N^3)$ (Golub and Van Loan, 1989). Using this approach, optimizing the regularization parameter

is of the order $O(KN^3 + RKN^2)$. Usually $R \ll N$ so that the order of the algorithm becomes $O(KN^3)$, which is independent of the number of $\lambda$'s $R$ that are tested. If $KN < M$, which is mostly the case for large datasets, this is in fact lower than the computational cost of Equation 3.2. Finding the optimal regularization parameter and calculating the optimal output weights is then of the order $O(N^2M)$. This means that optimizing the regularization parameter adds little to no computational cost to RR for large datasets.

Applying the eigenvalue decomposition to an ill-conditioned matrix results in numerical errors (Golub and Van Loan, 1989). This can be avoided by adding $\lambda_{max}\mathbf{I}$, with $\lambda_{max}$ the largest regularization parameter, to the covariance matrices (Golub and Van Loan, 1989) and later subtract it from the eigenvalues $\mathbf{D}_t$. Because of its simplicity, many of the mathematical toolboxes, such as the NumPy toolbox used in this work, use a variant of this technique by default.

## 3.2 Class-reweighted ridge regression

As mentioned in Chapter 2 RR has trouble finding the optimal separation between two classes if the dataset is unbalanced, such that there are more data points in one of the two classes. Because epileptic seizures are such rare events this is the case for seizure detection tasks. Class-reweighted RR (CRRR) (Toh, 2008) tries to solve this issue by rescaling the error for each class. It assumes that the error made on each class is independent of the number of data points in that class. This is achieved by dividing the mean squared error of each class separately by the number of elements in this class.

Figure 3.1 shows the difference of RR and CRRR applied to the example from Section 2.4. It shows that the results achieved using CRRR seem to give a slightly better separation between seizure and non-seizure samples for the first and second order polynomial expansion. Whether this technique retains its advantage when applied to RC will be discussed in the next chapter.

In mathematical notation, CRRR minimizes the following loss

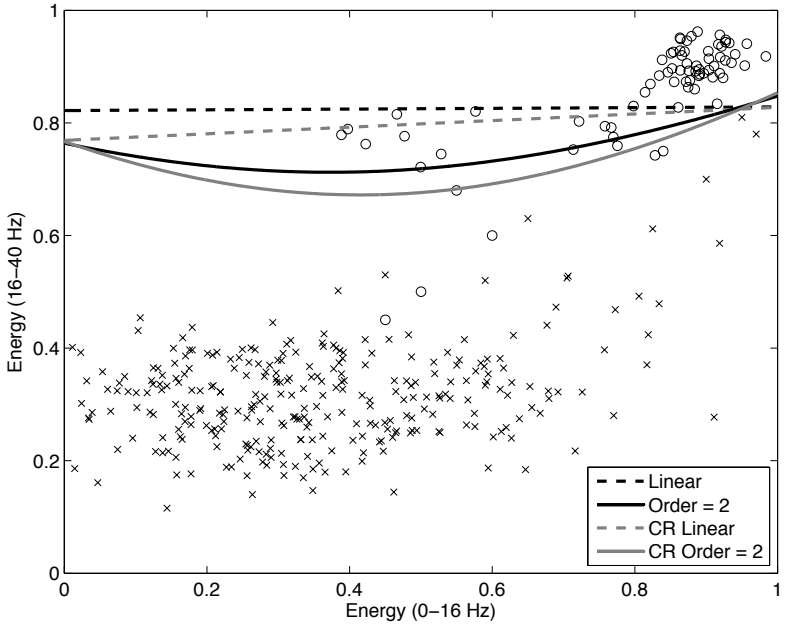


**Figure 3.1:** An example of a polynomial expansion to separate an epileptic seizure from normal EEG. Both the first and second order are shown for linear regression and class-reweighted linear regression. The threshold is optimized to have no false positives.

function as opposed to equation 3.1:

$$\begin{aligned} f_{loss} &= \frac{1}{n_{pos}}||(\mathbf{X}_t^+\mathbf{w}_t - \mathbf{y}_t^+)||^2 + \frac{1}{n_{neg}}||(\mathbf{X}_t^-\mathbf{w}_t - \mathbf{y}_t^-)||^2 + \lambda||\mathbf{w}_t||^2 \\ &= ||\mathbf{R}_t(\mathbf{X}_t\mathbf{w}_t - \mathbf{y}_t)||^2 + \lambda||\mathbf{w}_t||^2, \end{aligned}$$

with $\mathbf{R}_t$ a diagonal matrix that reweighs all positive examples to $\frac{1}{\sqrt{n_{t,pos}}}$ and all negative samples to $\frac{1}{\sqrt{n_{t,neg}}}$ with $n_{t,pos}$ and $n_{t,neg}$ the number of samples in the train set in the positive and negative class, respectively. Minimizing $f_{loss}$ results in the following:

$$W_{t,opt} = (\mathbf{X}_t^T\mathbf{R}_t^T\mathbf{R}_t\mathbf{X}_t + \lambda\mathbf{I})^{-1}\mathbf{X}_t^T\mathbf{R}_t^T\mathbf{R}_t\mathbf{y}_t.$$

To use CRRR in combination with the algorithm described in

the previous section, it suffices to compute covariance matrices $\mathbf{X}^T\mathbf{X}$, ${\mathbf{X_v}}^T\mathbf{X_v}$, $\mathbf{X}^T\mathbf{y}$, ${\mathbf{X_v}}^T\mathbf{Y_v}$ and ${\mathbf{Y_v}}^T\mathbf{Y_v}$ independently for each class. The positive and negative covariance matrices can be combined using the following formula:

$$\mathbf{A} = \frac{1}{n_{pos}}\mathbf{A}^+ + \frac{1}{n_{neg}}\mathbf{A}^-,$$

with $\mathbf{A}^+$ and $\mathbf{A}^-$ respectively the positive and negative covariance matrix. After computing the covariance matrices the rest of the previously described algorithms can be executed.

## 3.3 Feature selection algorithm

Feature selection (Guyon and Elisseeff, 2003) is a form of regularization where over-fitting is avoided by removing redundant or uninformative features. It follows the prior assumption that some of the inputs contain irrelevant data which is better ignored. Validating each possible combination of features is intractable for most common tasks. It scales exponentially for the number of inputs $N$ since the number of possible combinations equals $2^N - 1$. This means that for each additional feature the computational cost is almost doubled. For 3 inputs there are 7 possible combinations, for 4 there are 15, for 5 it is 31, and so on. It is not uncommon to have 100 or more inputs for which there are $10^{30}$ possible combinations that need to be tested. Since the computation time required to compute this for an average task is easily longer than a century on a modern day computer, a forward or backward approach is mostly used to find a near to optimal set of inputs. In forward feature selection, features are progressively added onto larger and larger subsets until no further performance increase is achieved. Backward feature selection on the other hand starts with a set containing all features. In each iteration all the remaining features are removed separately. The feature that, when removed, caused the largest decrease in validation error is eliminated from the subset. The algorithm is repeated until there is no further improvement. Forward algorithms are generally faster and result in fewer features but

backward algorithms achieve often a better performance because they tend to better preserve constructive relationships between seemingly irrelevant features (Guyon and Elisseeff, 2003).

Ridge regression (RR) doesn't automatically yield a sparse weight matrix as opposed to Lasso (Efron et al., 2004) or $L_1$ regularized regression. Many publications show however the advantages of RR (Dutoit et al., 2009; Ojeda et al., 2008; Pahikkala et al., 2010) or even linear regression (Miche et al., 2010) with sparse inputs. In Ojeda et al. (2008) and Miche et al. (2010) a fast algorithm for feature selection with small datasets was proposed. The algorithm presented in Pahikkala et al. (2010) also combines feature selection algorithm with the optimization of the regularization parameter. These algorithms have a computational complexity and memory use that scales quadratically with the size of the training set. This is however not suited for large datasets. Below an algorithm based on the eigen decomposition will be derived for regularized forward and backward feature selection (RFFS and RBFS), that is optimized for large datasets. Respectively, this is forward and backward feature selection combined with regularization parameter optimization into one algorithm.

### 3.3.1 Computational requirements of the naive implementation and covariance method

If there are $N$ inputs to the RR algorithm and if $N_s$ represent the number of selected features in a forward algorithm, the procedure of Section 3.1.1 needs to be repeated at maximum $NN_s$ times for a matrix with a dimensionality of at most $N_s \times M$, with $M$ the number of elements in the training set. For large datasets this becomes of the order $O(RKNN_s^3M)$. If $N_r$ represents the number of removed features using RBFS, the process of Section 3.1.1 needs to be repeated $NN_r$ times, so that the computational complexity for a naive RBFS is of the order $O(RKN^3N_rM)$.

RFFS and RBFS can also be achieved by performing the operation described in Section 3.1.2 on sub-matrices of the covariance matrices.

Simply removing the row and column $n$ for the $N \times N$ matrices and the row $n$ for the $N \times 1$ matrices, where $n$ is the feature to be removed, will already be more efficient than the naive implementation. This way selecting the best set of features and regularization parameter combined with training the output weights is an algorithm of the order $O(N^2M + RKNN_s^4)$ and $O(N^2M + RKN^4N_r)$ for the RFFS and RBFS respectively. The following sections will show how this can be further reduced. Let us start with RBFS since it is conceptually easier.

### 3.3.2 Backward feature selection

Because the $n$-th element on the diagonal of $\mathbf{A}_t$ is inversely proportional to the sensitivity of the output on the $n$-th input (Holland, 1973; Allen, 1974), one can remove a feature by setting the $n$-th diagonal element to $\infty$. If $\mathbf{u}$ is a vector containing zeros except for the $n$-th element which is equal to 1, we can compute the reduced matrix inverse using the Sherman-Morrison formula (Sherman and Morisson, 1950) as follows:

$$\begin{aligned}
\mathbf{A}_r^{-1} &= \lim_{\gamma\to\infty} (\mathbf{A}_t + \gamma\mathbf{u}\mathbf{u}^T)^{-1} \\
&= \lim_{\gamma\to\infty} \left( \mathbf{A}_t^{-1} - \frac{\mathbf{A}_t^{-1}\gamma\mathbf{u}\mathbf{u}^T\mathbf{A}_t^{-1}}{1+\gamma\mathbf{u}^T\mathbf{A}_t^{-1}\mathbf{u}} \right) \\
&= \mathbf{A}_t^{-1} - \lim_{\gamma\to\infty} \left( \frac{\gamma\mathbf{A}_t^{-1}\mathbf{u}\mathbf{u}^T\mathbf{A}_t^{-1}}{1+\gamma\mathbf{u}^T\mathbf{A}_t^{-1}\mathbf{u}} \right) \\
&= \mathbf{A}_t^{-1} - \frac{\mathbf{A}_t^{-1}\mathbf{u}\mathbf{u}^T\mathbf{A}_t^{-1}}{\mathbf{u}^T\mathbf{A}_t^{-1}\mathbf{u}}.
\end{aligned}$$

If we substitute the real and symmetric covariance matrix $\mathbf{A}_t$ for its eigen-decomposition (Parlett, 1980) in the previous equation we get:

$$\begin{aligned}
\mathbf{A}_r^{-1} &= \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T - \frac{\mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T\mathbf{u}\mathbf{u}^T\mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T}{\mathbf{u}^T\mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T\mathbf{u}} \\
&= \mathbf{C}\left( \mathbf{D}^{-1} - \frac{\mathbf{D}^{-1}\mathbf{C}_{(n,:)}^T\mathbf{C}_{(n,:)}\mathbf{D}^{-1}}{\mathbf{C}_{(n,:)}\mathbf{D}^{-1}\mathbf{C}_{(n,:)}^T} \right)\mathbf{C}^T,
\end{aligned}$$

where $\mathbf{C}_{(n,:)}$ denotes the $n$-th row of $\mathbf{C}$. If we introduce this in equations (3.6) and (3.7), using the same procedure as described in Section 3.1.3, we find that the output weights with a removed feature become:

$$\mathbf{w}_{Ctr} = \mathbf{w}_{Ct} - (\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{C}^T_{(n,:)} \frac{\mathbf{C}_{(n,:)}\mathbf{w}_{Ct}}{\mathbf{C}_{(n,:)}(\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{C}^T_{(n,:)}}.$$

Because $\mathbf{D}$ is a diagonal matrix and $\mathbf{w}_{Ct}$ and $\mathbf{C}^T_{(n,:)}$ have size $N \times 1$, this equation is again of the order $O(N)$. Testing the exclusion of 1 feature is thus of the same order of complexity as testing one ridge-regression parameter, $O(N^2)$. If this is tested for each of the $K$ validation sets, each of the $R$ regularization parameters and, in the worst case, each of the $N$ features, this becomes of the order $O(RKN^3)$. This is of higher order than the eigenvalue decomposition. We need to repeat this process $N_r + 1$ times, with $N_r$ the number of removed features, to find a near optimal set of features. The added complexity of the backward feature selection and regularization parameter optimization is thus of the order $O(RKN^3N_r)$. This results in a full algorithm with a complexity of $O(N^2M + RKN^3N_r)$.

### 3.3.3 Forward feature selection

When a feature is removed, the $n$-th row and column of the reduced covariance matrix $\mathbf{A}_r$, are all zero, with $n$ the index of the removed feature. When features are added we can thus start from the matrix $\mathbf{A}_r$. If we want to add a feature to this matrix we need to do the following $\mathbf{A}_e = \mathbf{A}_r + \mathbf{A}_a$, where $\mathbf{A}_a$ is a rank 2 matrix that contains all zeros except for the elements missing in $\mathbf{A}_r$ to create the covariance matrix with the added feature $\mathbf{A}_e$. Because $\mathbf{A}_a$ is symmetric and has rank 2 it can be decomposed in $\mathbf{A}_a = \mathbf{U}\mathbf{R}\mathbf{U}^T$, with $\mathbf{U}$ of size $(N_s + 1) \times 2$ and $\mathbf{R}$ of size $2 \times 2$, with $N_s$ the number of already selected features. If the last row and column of $\mathbf{A}_r$ corresponds to the missing features, the elements of $\mathbf{U}$ and $\mathbf{R}$ can be easily determined

as follows:

$$\mathbf{R} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$
$$\mathbf{U}^T = \begin{pmatrix} \mathbf{0} & 1 \\ \mathbf{A}_{(n,1:N_s)} & \frac{1}{2}\mathbf{A}_{(n,n)} \end{pmatrix},$$

with $\mathbf{A}_{(n,n)}$ the $n$'th diagonal element of $\mathbf{A}$ and $\mathbf{A}_{(n,1:N_s)}$ the elements corresponding to the already selected features on the $n$'th row of $\mathbf{A}$. Using the Sherman-Morrison-Woodbury formula (Golub and Van Loan, 1989) we can now determine the inverse of $\mathbf{A}_e$ as follows:

$$\begin{aligned}
\mathbf{A}_e^{-1} &= (\mathbf{A}_r + \mathbf{U}\mathbf{R}\mathbf{U}^T)^{-1} \\
&= \mathbf{A}_r^{-1} - \mathbf{A}_r^{-1}\mathbf{U}(\mathbf{R}^{-1} + \mathbf{U}^T\mathbf{A}_r^{-1}\mathbf{U})^{-1}\mathbf{U}^T\mathbf{A}_r^{-1} \\
&= \mathbf{C}(\mathbf{D}^{-1} - \mathbf{D}^{-1}\mathbf{C}^T\mathbf{U}(\mathbf{R}^{-1} + \mathbf{U}^T\mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T\mathbf{U})^{-1}\mathbf{U}^T\mathbf{C}\mathbf{D}^{-1})\mathbf{C}^T \\
&= \mathbf{C}(\mathbf{D}^{-1} - \mathbf{D}^{-1}\mathbf{C}_U^T(\mathbf{R}^{-1} + \mathbf{C}_U\mathbf{D}^{-1}\mathbf{C}_U^T)^{-1}\mathbf{C}_U\mathbf{D}^{-1})\mathbf{C}^T,
\end{aligned}$$

with the eigenvalue decomposition of $\mathbf{A}_r = \mathbf{C}\mathbf{D}\mathbf{C}^T$ and $\mathbf{C}_U = \mathbf{U}^T\mathbf{C}$. If we introduce this in equations (3.6) and (3.7) we find that the output weights with an added feature become:

$$\mathbf{w}_{Cte} = \mathbf{w}_{Ct} - (\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{C}_U^T(\mathbf{R}^{-1} + \mathbf{C}_U(\mathbf{D} + \lambda\mathbf{I})^{-1}\mathbf{C}_U^T)^{-1}\mathbf{C}_U\mathbf{w}_{Ct}.$$

Because $\mathbf{D}$ is a diagonal matrix, $\mathbf{w}_{Ct}$ has size $N \times 1$, $\mathbf{C}_U$ has size $N \times 2$ and $\mathbf{R}$ has size $2 \times 2$, this equation, when executed in the right order, is of the order $O(2(N_s + 1)) = O(N_s)$, with $N_s$ the number of selected features. Thus testing the addition of 1 feature or testing 1 regularization parameter is of the order $O(N_s^2)$. The eigenvalue decomposition for each validation set is of the order $O(KN_s^3)$. After that, one needs to test the addition of, at maximum, $N$ features and $R$ regularization parameters for each validation set, which is of the order $O(RKNN_s^2)$. To find the optimal set of features, this process needs to be repeated $N_s + 1$ times. Finding the optimal features and regularization parameter using forward feature selection is thus of the order $O(N^2M + KN_s^4 + RKNN_s^3) = O(N^2M + RKNN_s^3)$.

## 3.4 Bayesian relevance regression

Epileptic seizures differ a lot from patient to patient. Even the interictal EEG can differ a lot, especially the abnormal rhythmic non-seizure activity. During training a system tries to fit all the examples in the training set. Some of these examples might not be as relevant as others. Trying to fit all examples might cause the system to learn irrelevant seizure features. Bayesian relevance regression (BRR) scales the influence of each training example according to how statistically relevant it is with respect to the common model we want to train. It uses the prior assumption that the model will not be able to fit each example equally well. This helps to guarantee that the trained model does not overspecialise in detecting uncommon examples and is able to generalize better to unseen data.

BRR is similar to the automatic outlier detection technique presented in Ting et al. (2007). The following sections introduce the mathematical fundamentals of BRR. For more details on the mathematical derivations the reader is referred to Bishop (2006) and Coone (2011).

### 3.4.1 Relevance in a probabilistic setting

As shown in Section 2.2, linear regression tries to minimise the following loss function for the weights $\mathbf{w}$:

$$f_{loss}(\mathbf{w}) = \sum_{i=1}^{M} (y_i - \mathbf{w}^T \mathbf{x}_i)^2,$$

where $M$ is the number of elements in the training set and $y_i$ is the desired output for input vector $\mathbf{x}_i$. For every input vector the model makes an error $\epsilon_i$ so that the output can be determined as follows:

$$y_i = \mathbf{w}^T \mathbf{x}_i + \epsilon_i.$$

In a probabilistic setting we can now assume that this $\epsilon$ is an observation of Gaussian noise with zero mean and variance $\beta_k^{-1}$:

$$p(\epsilon_i|\beta_k) = \mathcal{N}(\epsilon_i|0, \beta_k^{-1}). \tag{3.8}$$

As opposed to (sparse) Bayesian linear regression (Bishop, 2006; Tipping, 2001), a different variance for each training set example $k$ is assumed. This results in a different variation on the model for each example. These examples can consist out of single data points or clusters of data points. In this work data points from the same time-series are clustered since there is a correlation between consecutive reservoir states. For seizure detection this means that the model assumes that each EEG time-series will not be fitted with the same accuracy. The $\beta_k$ parameters could thus be used to set the influence of each these examples on the model and can be scaled according to their relevance.

For $y$, Equation 3.8 results in the following likelihood function :

$$p(y|\mathbf{w}, \mathbf{x}, \beta_k) = \mathcal{N}(y|\mathbf{w}^T\mathbf{x}, \beta_k^{-1}).$$

The joint likelihood for all training examples is obtained by multiplying the likelihoods of the individual examples. For the full training set this becomes:

$$\begin{aligned} p(\mathbf{y}|\mathbf{w}, \mathbf{X}, \boldsymbol{\beta}) &= \prod_{k=1}^{K}\prod_{i=1}^{M_k} p(y_{ki}|\mathbf{w}, \mathbf{x}_{ki}, \beta_k) \\ &= \prod_{k=1}^{K}\prod_{i=1}^{M_k} \mathcal{N}(y_{ki}|\mathbf{w}^T\mathbf{x}_{ki}, \beta_k^{-1}) \\ &= \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{w}, \boldsymbol{\beta}^{-1}), \end{aligned}$$

with $K$ the number of training examples, $M_k$ the number of data points within each training example and $\boldsymbol{\beta}$ a diagonal matrix with on its diagonal the corresponding $\beta_k$ for each data point. When we maximize this equation we get the maximum likelihood estimate for $\mathbf{w}$ which is equivalent to optimizing the loss function. For the remainder of this chapter we omit $\mathbf{x}$ and $\mathbf{X}$ from the probability density function parameters to avoid cluttered notation.

### 3.4.2 Probabilistic regularization

The maximum likelihood results in a system that fits the training data very well. However, to avoid over-fitting it makes more sense to maximize how certain we are about the weights $\mathbf{w}$. Given the data we want to find the most likely set of parameters and not the other way around. Using Bayes' rule, we can write $p(\mathbf{w}|\mathbf{y})$ as a function of $p(\mathbf{y}|\mathbf{w})$ and $p(\mathbf{w})$:

$$\begin{aligned} p(\mathbf{w}|\mathbf{y}) &= \frac{p(\mathbf{y}|\mathbf{w})p(\mathbf{w})}{p(\mathbf{y})} \\ &= \frac{p(\mathbf{y}|\mathbf{w})p(\mathbf{w})}{\int p(\mathbf{y}|\mathbf{w})p(\mathbf{w})d\mathbf{w}} \end{aligned} \tag{3.9}$$

Maximizing this equation results in the so called maximum a posteriori (MAP) estimate. The probability of $p(\mathbf{w})$ is called the prior and $p(\mathbf{w}|\mathbf{y})$ the posterior distribution of the set of parameters. We assume the prior to be Gaussian with a mean $\boldsymbol{\mu}$ and a variance $\alpha^{-1}$:

$$p(\mathbf{w}|\alpha) = \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \alpha^{-1}).$$

This $\boldsymbol{\mu}$ is a prior assumption on the weights and is usually set to zero. Applying the general result to compute the marginal distribution of a combination of two multivariate Gaussians, derived in (Bishop, 2006), to Equation 3.9, results in the following for the MAP:

$$\begin{aligned} p(\mathbf{w}|\mathbf{y}) &= \mathcal{N}(\mathbf{w}|\mathbf{m}_n, \mathbf{S}_n) \\ \mathbf{m}_n &= \mathbf{S}_n(\alpha\mathbf{I}\boldsymbol{\mu} + \beta\mathbf{X}^T\mathbf{y}) \\ \mathbf{S}_n^{-1} &= \alpha\mathbf{I} + \beta\mathbf{X}^T\mathbf{X}. \end{aligned}$$

Here $\mathbf{m}_n$ is the expected value of the MAP and represents the most probable weights. This is very similar to what was found for RR. The $\alpha$ parameter can be compared to the regularization parameter in RR when $\boldsymbol{\mu}$ is considered zero. If the prior $\boldsymbol{\mu}$ is not set to zero, deviation from these prior weights is punished as opposed to the size of the weights.

### 3.4.3 Hyper-parameter optimization

Previous sections have shown how relevance and regularization can be integrated in a probabilistic setting, but the most important part is how to optimize the hyper parameters. Given a new data point $\check{\mathbf{x}}$, we want to be as certain as possible about the prediction $\check{y}$. According to Bishop (2006), this can be written as follows:

$$p(\check{y}|\mathbf{y}, \alpha, \boldsymbol{\beta}) = \mathcal{N}(\check{y}|\mathbf{m}_n^T\check{\mathbf{x}}, \boldsymbol{\beta} + \check{\mathbf{x}}^T\mathbf{S}_n\check{\mathbf{x}}).$$

This model is still dependent on the hyper-parameters, $\alpha$ and $\boldsymbol{\beta}$, and $\mathbf{w}$. We can marginalise by integrating over these variables, so that we get the following predictive distribution which is independent of these variables:

$$p(\check{y}|\mathbf{y}) = \int\int\int p(\check{y}|\mathbf{w}, \boldsymbol{\beta})p(\mathbf{w}|\mathbf{y}, \alpha, \boldsymbol{\beta})p(\alpha, \boldsymbol{\beta}|\mathbf{y})d\mathbf{w}d\alpha d\boldsymbol{\beta}$$

The problem with this integration however, is that it is intractable. A good estimate can be found by applying the evidence approximation. This approach assumes that the posterior distribution over the hyper-parameters $p(\alpha, \boldsymbol{\beta}|\mathbf{y})$ is sharply peaked. Under this assumption, an integration over these hyper-parameters is no longer necessary, given good estimates $\hat{\alpha}$ and $\hat{\boldsymbol{\beta}}$ (Bishop, 2006). The predictive distribution now becomes:

$$p(\check{y}|\mathbf{y}) = \int p(\check{y}|\mathbf{w}, \hat{\boldsymbol{\beta}})p(\mathbf{w}|\mathbf{y}, \hat{\alpha}, \hat{\boldsymbol{\beta}})d\mathbf{w}$$

The optimal $\alpha$ and $\boldsymbol{\beta}$ can be found by maximizing the evidence:

$$p(\mathbf{y}|\alpha, \boldsymbol{\beta}) = \int p(\mathbf{y}|\mathbf{w}, \boldsymbol{\beta})p(\mathbf{w}|\alpha)d\mathbf{w}.$$

Let us now look at each part of the evidence function separately:

$$\begin{aligned} p(\mathbf{y}|\mathbf{X}\mathbf{w}, \boldsymbol{\beta}) &= \mathcal{N}(\mathbf{y}|\mathbf{X}\mathbf{w}, \boldsymbol{\beta}^{-1}) \\ &= \left(\frac{1}{2\pi}\right)^{\frac{M}{2}} |\boldsymbol{\beta}|^{\frac{1}{2}} \exp\left(-\frac{1}{2}(\mathbf{y}-\mathbf{X}\mathbf{w})^T\boldsymbol{\beta}(\mathbf{y}-\mathbf{X}\mathbf{w})\right) \\ p(\mathbf{w}|\alpha) &= \mathcal{N}(\mathbf{w}|\boldsymbol{\mu}, \alpha^{-1}) \\ &= \left(\frac{1}{2\pi}\right)^{\frac{N}{2}} |\alpha|^{\frac{N}{2}} \exp\left(-\frac{\alpha}{2}(\mathbf{w}-\boldsymbol{\mu})^T(\mathbf{w}-\boldsymbol{\mu})\right), \end{aligned}$$

with $N$ the number inputs and $M$ the number of data points. Combining these equations results in the following for the evidence function:

$$p(\mathbf{y}|\alpha, \boldsymbol{\beta}) = \left(\frac{1}{2\pi}\right)^{\frac{N+M}{2}} |\alpha|^{\frac{N}{2}} |\boldsymbol{\beta}|^{\frac{1}{2}} |\mathbf{S}_n|^{\frac{1}{2}} \exp\left(-E(\mathbf{w})\right),$$

with $E(\mathbf{w}) = \frac{1}{2}(\mathbf{y}-\mathbf{X}\mathbf{w})^T\boldsymbol{\beta}(\mathbf{y}-\mathbf{X}\mathbf{w}) + \frac{\alpha}{2}(\mathbf{w}-\boldsymbol{\mu})^T(\mathbf{w}-\boldsymbol{\mu})$. Maximising this evidence function is equivalent to maximising the logarithm of this evidence function. Taking the derivative of $\ln(p(\mathbf{y}|\alpha, \boldsymbol{\beta}))$ to $\alpha$ and $\beta_k$ and setting to zero gives the following:

$$\begin{aligned} \frac{\partial}{\partial\alpha}\ln(p(\mathbf{y}|\alpha, \boldsymbol{\beta})) &= \frac{N}{2\alpha} - \frac{1}{2}\mathrm{tr}(\mathbf{S}_n) - \frac{1}{2}||\mathbf{w}-\boldsymbol{\mu}||^2 = 0 \\ \frac{\partial}{\partial\beta_k}\ln(p(\mathbf{y}|\alpha, \boldsymbol{\beta})) &= \frac{M_k}{2\beta_k} - \frac{1}{2}\mathrm{tr}(\mathbf{S}_n(\mathbf{X}_k^T\mathbf{X}_k)) - \frac{1}{2}||\mathbf{y}_k - \mathbf{X}\mathbf{w}||^2 = 0, \end{aligned}$$

where $\mathrm{tr}(\mathbf{A})$ is the trace of $\mathbf{A}$, the sum of the elements on the diagonal. These equations yield:

$$\begin{aligned} \alpha &= \frac{N}{\mathrm{tr}(\mathbf{S}_n) + ||\mathbf{w}-\boldsymbol{\mu}||^2} \\ \beta_k &= \frac{M_k}{\mathrm{tr}(\mathbf{S}_n\mathbf{X}_k^T\mathbf{X}_k) + ||\mathbf{y}_k - \mathbf{X}_k\mathbf{w}||^2}. \end{aligned}$$

As stated in the previous section, $\mathbf{S}_n$ depends on the $\alpha$ and $\beta_k$s we try to find. However, iteratively calculating $\mathbf{S}_n$ and $\mathbf{w}$ followed by $\alpha$ and each $\beta_k$ converges to the optimal values (Ting et al., 2007; Bishop, 2006; Tipping, 2001).

As initial values $\alpha = 0$ and $\beta_k = 1$ are typically used and $\boldsymbol{\mu}$ is set to zero. The covariance technique from Section 3.1.2 can be applied to

$\alpha = 0$
**for** k in [1,K] **do**
  $\beta_k = 1$
  $\mathbf{A}_k = \mathbf{X}_k^T \mathbf{X}_k$
  $\mathbf{b}_k = \mathbf{X}_k^T \mathbf{y}_k$
  $c_k = \mathbf{y}_k^T \mathbf{y}_k$
  $M_k = \text{length}(\mathbf{y}_k)$
**end for**
**while not** converged **do**
  $\mathbf{S}_n^{-1} = \alpha \mathbf{I} + \sum_{k=1}^{K} \beta_k \mathbf{A}_k$
  $\mathbf{w} = \mathbf{S}_n \sum_{k=1}^{K} \beta_k \mathbf{b}_k$
  $\alpha = N / \Big( \text{tr}(\mathbf{S}_n) + \mathbf{w}^T \mathbf{w} \Big)$
  **for** k in [1,K] **do**
    $\beta_k = M_k / \Big( \text{tr}(\mathbf{S}_n \mathbf{A}_k) + \mathbf{w}^T (\mathbf{A}_k \mathbf{w} - 2\mathbf{b}_k) + c_k \Big)$
  **end for**
**end while**

**Alg. 3.1:** Bayesian relevance regression

reduce the computational cost. This results in the update algorithm shown in Algorithm 3.1, with a computational cost of the order $O(N^3)$ for each iteration. On the tasks used in this work convergence is usually reached after 10 to 20 iterations.

## 3.5 Conclusion

In this chapter a newly derived optimization algorithm for large datasets was proposed to optimize the regularisation parameter. This technique is based on the eigenvalue decomposition. It was extended to be applied in combination with class reweighted RR and feature selection. In Section 3.4 a new regularization method, Bayesian relevance regression, was derived which scales the influence of each example according to its relevance to the model.

If $M$ is the number of data points in the training set and $N$ the number of input features, each of these methods require a compu-

**Table 3.1:** The computational cost to optimize the regularization parameter and train the readout weights for the different methods discussed in this chapter, given large datasets are used: $N \ll M$ and $R \ll N$. Here $N$ equals the number of input features (a few hundred to a few thousand), $M$ the number of data points in the training set (up to a few million or more), $R$ the number of regularization parameters (around 50), $K$ the number of validations sets (around 10), $N_s$ the number of selected features in forward feature selection (maximum $N$), $N_r$ the number of removed features in backward feature selection (maximum $N-1$) and $I$ the number of iterations for the BRR algorithm (around 20).

| **Methods** | naive | covariance | eigen |
|---|---|---|---|
| RR | $RKN^2M$ | $N^2M + RKN^3$ | $N^2M$ |
| CRRR | $RKN^2M$ | $N^2M + RKN^3$ | $N^2M$ |
| RFFS | $RKNN_s^3M$ | $N^2M + RKNN_s^4$ | $N^2M + RKNN_s^3$ |
| RBFS | $RKN^3N_rM$ | $N^2M + RKN^4N_r$ | $N^2M + RKN^3N_r$ |
| BRR | $IKN^2M$ | $N^2M + IKN^3$ | n/a |

tational cost of the order $O(N^2M)$ to compute the readout weights if the regularisation parameter(s) and/or selected input features are known. In Table 3.1 the computational cost is shown to optimize the parameters and/or feature set and to train the readout weights, for the methods from literature and the methods introduced in this chapter. For the RR-based techniques it shows that the theoretical computational cost is significantly less for the introduced eigen-decomposition-based methods. For RR and class reweighted RR it is even equal to the cost to compute the weights. The naive and covariance methods form literature have a significantly higher computational cost. The Bayesian relevance regression regularisation technique, introduced in this chapter, has a higher computational cost compared to RR and class reweighted RR, but is significantly less computationally expensive than the regularised forward and backward feature selection algorithms. For this method only the covariance method could be applied

to reduce the computational cost. The performance of these algorithms will be evaluated in the next chapter.

# 4 Seizure detection in animal models

For epilepsy research animal models are applied to evaluate the therapeutic efficacy of anti-epileptic treatment (Dedeurwaerdere, 2005). In order to validate these treatments the number of seizures and their duration need to be determined. This results in many hours of tedious EEG review and analysis. Automated seizure detection decreases the workload and may also be more reliable compared to hours of visual analysis. Real-time seizure detection can be incorporated in a so called closed-loop system (Stein et al., 2000) that allows immediate triggering of an intervention at the time of seizure occurrence such as: fast working anti-epileptic drugs, deep brain stimulation (Waterschoot et al., 2006; Wyckhuys et al., 2010), vagus nerve stimulation (Boon et al., 2001), etc. The seizure detection algorithms presented in this chapter were published in Buteneers et al. (2010) and Buteneers et al. (2012b).

## 4.1 Materials

The EEG used in this chapter contains absence seizures from genetic absence epilepsy rats from Strasbourg (GAERS) and limbic seizures from post status epilepticus (PSE) rats. It originates from experiments to evaluate new anti-epileptic treatment therapies. The data has been recorded with a custom-built amplifier at a sample rate of 200 Hz or higher. For consistency, data sampled at a higher frequency

was subsampled, so that the sample rate was 200 Hz for all animals. After recording the data was evaluated by experienced encephalographers.

The complete dataset consists of 454 hours of data from 23 GAERS and 2083 hours of data from 22 PSE rats (see Table 4.1). For GAERS, the training set, 5.75 hours in total, consists of the first 15 minutes of EEG per rat which contained at least 90 seconds of ictal EEG. The training set for the PSE data, 44 hours of data, consists of the first 10 seizures of each rat in the dataset combined with 10 minutes inter-ictal EEG equally distributed over pre-ictal and post-ictal EEG. The rest of the data following the training data was used for testing. In the following two sections a more detailed description is given on the origin of the datasets.

### 4.1.1 Genetic absence epilepsy rats from Strasbourg

GAERS are a strain of Wistar rats that all exhibit spontaneous absence seizures characterized by a sudden unresponsiveness to environmental stimuli and cessation of ongoing activity (Marescaux et al., 1992). These absence seizures, which are displayed as synchronous spike and wave discharges (SWDs) on the EEG, occur mostly when the animal is in a state of quiet wakefulness. They are rare during periods of active arousal and sleep. The number of seizures and their duration increase with age, until they reach a maximum at about 6 months. The EEG of SWDs shows a fundamental frequency in the range of 7 to 12 Hz and several harmonics (see Figure 4.1), an amplitude varying from 300 to 1000 μV and a duration from 0.5 to 120 s.

Dataset A was made during a study to evaluate the effect of acute and non-acute high (130 Hz) and middle high (60 Hz) frequency deep brain stimulation on the occurrence of SWDs (Waterschoot et al., 2006). The rats from dataset B were part of a study to evaluate the effect of long-term vagus nerve stimulation.

All EEG fragments were visually reviewed, the data contaminated with stimulation artefacts was removed, one EEG channel was selected and the SWDs with a minimum seizure length of 0.5 s were marked by

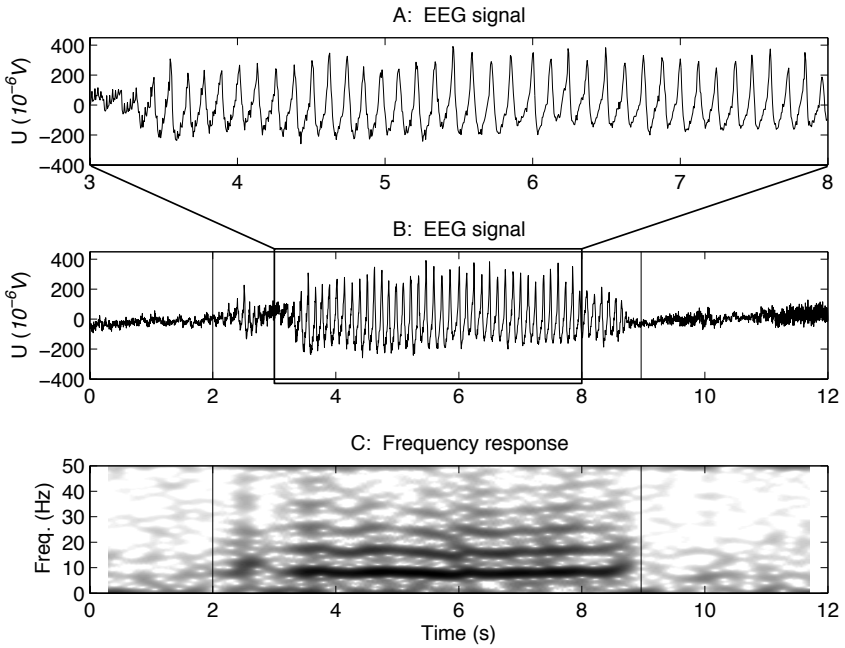


**Figure 4.1:** An example of a spike and wave discharge (SWD) caused by an absence seizure in genetic absence epilepsy rats from Strasbourg. In (A) and (B) the EEG signal of one intra-cranial channel is shown, where (A) is a magnified version of the marked area in (B). The seizure starts at time = 2 s and stops at time = 8.9 s. (C) shows the spectrogram of the EEG signal with a Hamming-window of 128 samples and an overlap of 120 samples.

an experienced encephalographer. These annotations were used as the 'gold standard' in this study. From the first study, 64.5 hours of single-channel depth EEG-data recorded in the anterodorsal thalamus from 12 different rats was used for dataset A. The 3468 seizures made up 23% of the total time and lasted on average 15 seconds. The second study yielded 390 hours of single-channel scalp EEG-data recorded over the frontoparietal cortex from 11 rats for dataset B. A total number of 6183 seizures made up 4.5% of the data and lasted, on

**Table 4.1:** The number of animals, total length of the dataset, the number of seizures and the average length of the seizures for each of the datasets.

| **GAERS** | animals | hours | seizures | length |
|---|---|---|---|---|
| A | 12 | 64.5 | 3468 | 15s |
| B | 11 | 390 | 6138 | 10s |

| **PSE** | animals | hours | seizures | length |
|---|---|---|---|---|
| C | 11 | 913 | 1541 | 54s |
| D | 7 | 1105 | 1374 | 42s |
| E | 4 | 69 | 113 | 51s |

average, 10 seconds. Each of the seizures lasted between 0.5 and 110 s.

### 4.1.2 Post status epilepticus rats

Kainic acid is a potent central nervous stimulant, isolated from the seaweed digenea simplex. This excitotoxic product is an agonist of a subclass of ionotropic glutamate receptors. A systemic injection in healthy rats triggers a cascade of molecular and cellular events eventually leading to status epilepticus, followed by a period of gradual increase in seizure frequency, which eventually stabilizes. Finally, rats display spontaneous, secondary generalized limbic seizures which resemble those seen in temporal-lobe epilepsy patients (Baraban, 2009).

During annotation, spontaneous seizures were recognized by their large amplitude (more than 3 times baseline amplitude) high-frequency EEG activity ($\geq$ 5 Hz), with characteristic high temporal correlation and progression of spike frequency. Figure 4.2 shows an example of a limbic seizure.

Dataset C was made during a study to evaluate the effect of long-term high frequency (130 Hz) and Poisson distributed high frequency (on average 130 Hz) deep brain stimulation on the occurrence of limbic

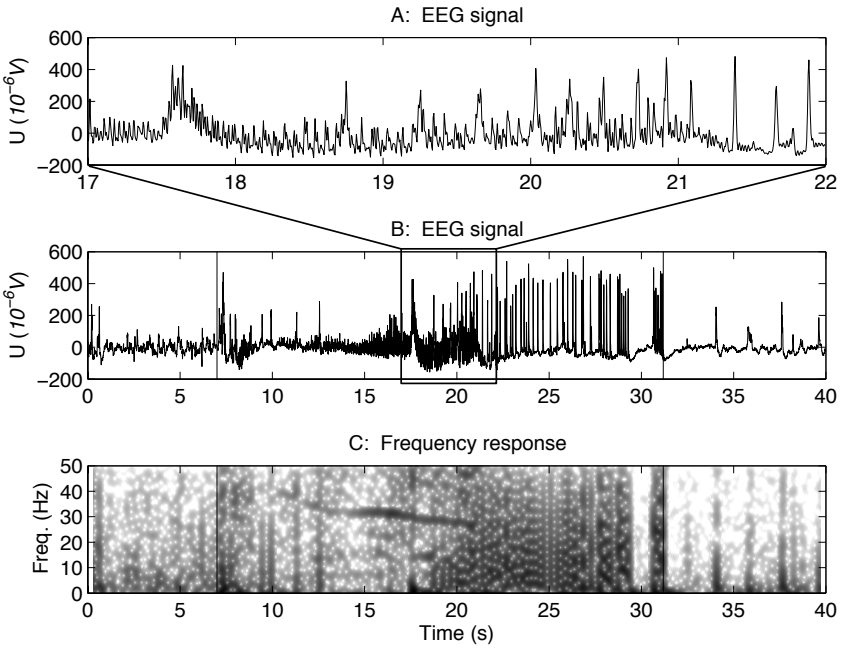


**Figure 4.2:** An example of a limbic seizure in a post status epilepticus rat. In (A) and (B) the EEG signal is shown, where (A) is a magnified version of the marked area in (B). The seizure starts at time = 7 s and stops at time = 31 s. (C) shows the spectrogram of the EEG signal with a Hamming-window of 128 samples and an overlap of 120 samples.

seizures (Wyckhuys et al., 2010). An experienced encephalographer evaluated all EEG fragments visually and marked all present seizures in dataset C. This resulted in 913 hours of four channel EEG from 11 different rats. Approximately 2.5% of this data consisted of 1541 seizures which have a duration of 9 to 240 seconds with an average of 54 seconds. In five animals deep brain stimulation was applied. This subset (C*) contained some episodes of EEG contaminated with stimulation artefacts (in the rest of this work referred to as $C^*_{stim}$). An example of a stimulation artefact is shown in Figure 4.3.

Study D compared the therapeutic effect of deep brain stimula-

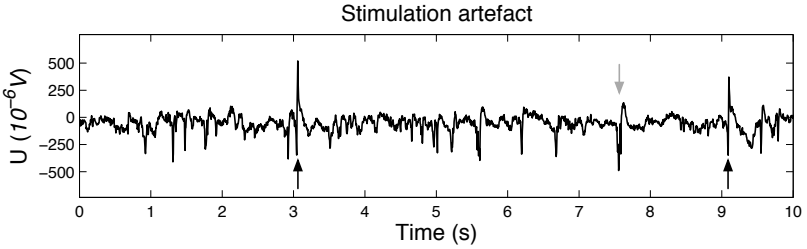


**Figure 4.3:** An example of stimulation artefacts caused by stimulation at 130 Hz. Because of the low sample rate (200 Hz), the stimulation is above the Nyquist frequency of 100 Hz. Therefore it is only visible as spikes in the EEG for example at time = 3.05 s and 9.1 s (up arrows). These spikes are somewhat similar to the many epileptic spikes as for example at time = 7.6 s (down arrow).

tion in the hippocampus and midline thalamic nuclei. Both experimental therapies were evaluated for their effect on the frequency of spontaneous seizures in the PSE model. The EEG fragments were visually evaluated and annotated. This resulted in 1105 hours of EEG from 7 different rats without simulation artefacts. 1374 seizures were recorded in total, which lasted on average 42 seconds or between 12 and 220 seconds and represent 1.4% of the data.

In study E the effect of introducing stem cells from foetal mice brains in the epileptogenic areas was studied on the occurrence of limbic seizures. All EEG fragments were visually evaluated and all seizures were marked. From this study 69 hours of 4 channel EEG from 4 different rats was used. Dataset E contained 113 seizures that were located in about 2.5% of the data and lasted 23 to 360 seconds with an average of 51 seconds.

Datasets C, D and E consist of 4 channel hippocampal EEG with a referential montage for each rat. From these 4 channels, one was chosen with visually the most significant difference between ictal and inter-ictal EEG based on the first 4 seizures.

## 4.2 Evaluation measures

The gold standard used to compare the different detection methods is the scoring by experienced encephalographers. For all animals the number of false positives and false negatives per seizure (FPPS and FNPS) are measured, together with the detection delay $\Delta_{delay}$ (in seconds). This delay is only determined for correctly detected seizures and includes the time required to perform preprocessing. As a lower bound, the first inter-ictal sample after the previous seizure is used and as an upper bound, the last marked sample of the to be detected seizure.

Each of the error measures is calculated for each animal individually. Then the mean and standard deviation is calculated over all the animals which is used for comparison. Each animal thus has the same contribution to the results, independent of the amount of data that was recorded for this animal. In the tables below the standard deviation is usually represented by a number between rounded brackets.

## 4.3 Methods from literature

Many epileptic seizure detection methods for animal models have been developed. Some were designed to aid in the marking of EEG, others to trigger anti-epileptic treatment in real-time. The algorithms that haven been shown to achieve the best performance in Van Hese et al. (2009) and Buteneers et al. (2010) are discussed below.

### 4.3.1 Adapted Osorio-Frei algorithm

The original Osorio-Frei method (Osorio et al., 1998) (OFA) is in fact the most frequently cited seizure detection method for human seizures. It owes its popularity to both its simplicity and its effectiveness for seizure detection. Although it was designed for human iEEG, it relies on a feature that is shared with epileptic seizures from rats and has been previously applied on rat data in Van Hese et al. (2003, 2009)

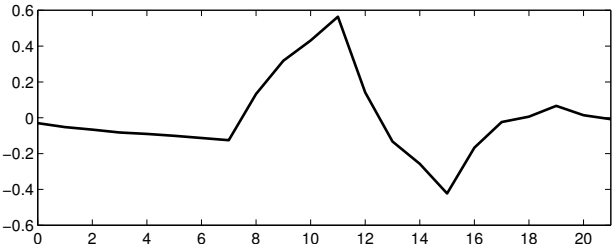


**Figure 4.4:** The coefficients of the level 3 Daubechies 4 wavelet filter.

and Buteneers et al. (2010). An extension to this algorithm, the adapted Osorio-Frei algorithm (AOFA) (Haas et al., 2007), extends the original algorithm with multiple features. From these features, the best feature is selected. It is compared with background EEG and if a certain threshold is exceeded a seizure is detected. The AOFA method has been previously applied on rat data in Buteneers et al. (2012b).

The technique starts by convolving the EEG signal with the signal shown in Figure 4.4. This waveform is based on the the Daubechies 4 level-3 wavelet coefficients. It is very similar to a spike and wave discharge and is used as filter coefficients to process the data. For a signal sampled at 240 Hz the output $y$ at time $k$ is calculated as follows:

$$y[k] = \sum_{i=0}^{21} b[i]x[k-i],$$

where $x[k]$ is the $k$-th input sample and $b[i]$ is the $i$-th filter coefficient. For an EEG signal sampled at 240 Hz, this wavelet filter has a pass band of 5 to 45 Hz. From the resulting signal the foreground signal is computed using a window of width $T_1$ as follows:

$$FG[k] = \text{median}\left\{y^2[k], y^2[k-1], ..., y^2[k-(T_1-1)]\right\}.$$

Here the median, as opposed to the average, is used since it is insensitive to outliers (Gallagher Jr and Wise, 1981). These outliers often occur in EEG due to measurement artefacts caused by electrical noise (Teplan, 2002). For human data $T_1$ is set to 480 samples or a width of 2 seconds. This foreground signal is rescaled using a back-

ground signal. It is basically a low pass filtering of the median filtered foreground signal and is computed as follows:

$$BG[k] = \begin{cases} (1-\lambda)\text{median}\{FG[k], ..., FG[k-(T_2-1)s]\} & \textbf{if k=ps,} \\ \quad + \lambda BG[k-1] & \\ BG[k-1] & \textbf{if } p(s-1) \leq k < ps, \end{cases}$$

where $p = 0, 1, 2, ..., s$ and $s = \frac{T_1}{4}$, so that only 1 out of 4 foreground samples is used to reduce the computational cost. $T_2$ is the width of the overlap window which is set to 480 or 4 minutes for human data and $\lambda = 0.999807$ which corresponds to a 30 minute half-life when the previous parameter values are used. This means that the influence of the median filtered foreground signal on the background scaling factor is halved every past 30 minutes. Next, a dimensionless ratio $R[k]$ is computed as follows:

$$R[k] = \max_{1 \leq n \leq N} \left\{ \frac{FG[k]^{(n)}}{BG[k]^{(n)}} \right\},$$

where $N$ denotes the number of channels. If this ratio $R[k]$ is above a threshold $\delta_{OF}$, sample $k$ can be considered part of a seizure. To reduce the number of false positives, an extra parameter $T_s$ is used to determine the minimum number of samples a seizure has to last. After there are more than $T_s$ neighbouring samples above $\delta_{OF}$, these samples are considered part of a seizure and thus a seizure is detected. Because of this $T_s$ is the parameter that influences the detection delay most significantly. In Osorio et al. (1998) the threshold $\delta_{OF}$ was found to be optimally set to 22 with $T_s = 0.84$ s to achieve the best results on the test data. Using these values the average detection delay was 2.1 s while detecting no false positives and missing none of the seizures. Since these values have been optimized on the test set the performance achieved in Osorio et al. (1998) is not necessarily an indication of good performance on other data sets but Osorio et al. (2002) reported similar performance for the sensitivity and detection delay albeit with a higher number of FPPS on a different dataset. The exact number can not be extracted from the paper but can be estimated at about 0.2 FPPS.

In the adapted version of the OFA algorithm (Haas et al., 2007), the best feature, or in this case filter, is selected using the so called seizure to non-seizure ratio (SNSR):

$$\text{SNSR} = \frac{P_p\{(\text{filtered seizure signal})^2\}}{P_p\{(\text{filtered non-seizure signal})^2\}},$$

where $P_p\{S\}$ represents the p-th percentile of the set $S$. Note that the 50-th percentile is equivalent to the median. To select the best feature this measure is evaluated for different features and percentiles. The feature-percentile combination with the highest SNSR is chosen and thus used for seizure detection. For more details on the features used we refer to Haas et al. (2007).

To optimize the algorithm for rats several adaptations were applied. A sample rate of 200 Hz was used as opposed to 240Hz. For both seizures the *ratio eigenfilter* feature (Haas et al., 2007) was selected using the SNSR. This feature is in fact a finite impulse response filter that is specifically designed to get the highest SNSR. However, during cross-validation and experiments on the test set, the results showed that this filter seemed to over-fit on the training data. The wavelet feature of the original OFA on the other hand, performed best in cross-validation experiments on the training set and is therefore used in the experiments below. The following parameters were also optimized during training: $T_1$, $T_2$, $T_s$ and $\delta_{OF}$.

### 4.3.2 The Van Hese algorithm

In Van Hese et al. (2009) an off-line spike and wave discharge (SWD) detection method to mark the EEG of GAERS was presented which exploits the fact that SWDs are quasi-periodic signals. They have a fundamental frequency from 7 to 12 Hz and several harmonics (see Figure 4.1). In a first step, the short term fourier transform is applied to non-overlapping and zero-padded intervals, which results in a spectrogram. In a second step, the background spectrum is determined for each of the frequency components as proposed in Stahl et al. (2000). In a following step, called the harmonic analysis, spectral peaks, higher than the background spectrum, are used to determine

**Table 4.2:** The FPPS, FNPS, detection delay and variance on the detection delay for the methods by Osorio et al. and Van Hese et al. tested on the datasets in this work. For the method by Van Hese et al. no detection delay was recorded since it was designed for off-line seizure marking of GAERS iEEG.

| **GAERS** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| AOFA | 1.4 (3.0) | 0.18 (0.20) | 2.5 (0.6) |
| Van Hese | 3.5 (3.2) | 0.11 (0.07) | n/a |

| **PSE** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| AOFA | 3.4 (3.8) | 0.032 (0.067) | 20 (4) |

the fundamental frequency and the harmonics at approximately 2, 3, or more times that frequency. If these signals contain an energy above a certain threshold, the interval is considered part of an SWD. For a more detailed explanation we refer to Van Hese et al. (2009). In this work the implementation of the authors was used and during training only the threshold was optimized.

### 4.3.3 Experiments

In order to evaluate the overall detection performance, each method was trained on the complete training set and tested on the complete test set. Table 4.2 presents the results for the different methods. Because the method by Van Hese uses features that are specific for SWDs it has not been applied on the data of the PSE rats.

The AOFA method falsely detects 1.4 SWDs and 3.4 limbic seizures for every true seizure while missing approximately 20% and 3% of the seizures respectively. As shown in Table 4.2 the method by Van Hese et al. has more than double the amount of false positives compared to the AOFA method and misses only slightly fewer SWDs.

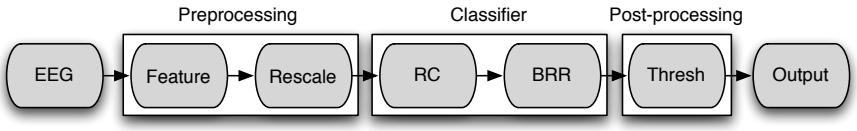


**Figure 4.5:** A schematic representation of the detection method presented in this work.

## 4.4 Reservoir computing

The ML based detection method using RC consists of three parts: a preprocessing stage where features get extracted from the EEG, a classification stage which is based on RC and some form of linear readout for classification, and a post-processing stage where two thresholds are applied. A schematic representation is shown in Figure 4.5. The next 3 sections discuss each of these stages with their possible design choices. Section 4.4.4 discusses the test results achieved with the different design choices.

### 4.4.1 Pre-processing

In the pre-processing stage the features, selected using ranked forward feature selection, are extracted. These features are first rescaled and used as input for the classifier.

#### Input feature selection and extraction

The relevant features from the EEG (Päivinen et al., 2005; Haas et al., 2007; Costa et al., 2008) are selected during training with a ranked forward feature selection algorithm. To create an optimal set of features, they are first ranked with respect to the balanced error rate (BER) (see Section 4.4.3) on the training set using cross-validation. The best feature is added to the set of features $FS$. Next the second best feature is added to this set. If this combination reduces the validation error this feature remains in the set. If not, the feature is removed from the set. This is repeated for all the features in

```
ranked_features = rank(features)
FS = ranked_features[1]
for i = 2 to length(ranked_features) do
    set = {FS, ranked_features[i]}
    if error(set) < error(FS) then
        FS = set
    end if
end for
```

**Alg. 4.1:** Forward feature selection

the list until an optimal set of features is found. The pseudo-code of this algorithm is shown in Algorithm 4.1. Usually forward feature selection algorithms do not rank the features but evaluate the added value of each of the features in each step. In order to reduce the computational cost however, a ranking method was added. This results in an algorithm that has a computational complexity of the order $O(2N) = O(N)$, multiplied by the complexity to evaluate the performance of one feature. This is equal to the best case scenario for a conventional forward feature selection. The worst case scenario for conventional forward feature selection is however $\frac{N}{2}$ times higher, which results in an algorithm of the complexity of $O(N^2)$ times the complexity to evaluate one feature.

The EEG features used were: a filter bank of Butterworth filters ranging from 1 to 30 Hz with a bandwidth of 2 Hz, a set of Daubechies 4 wavelet filters (level 2 to 6), the first derivative of the EEG signal, the energy of the signal and the energy in the theta (4 to 8 Hz), alpha (8 to 12 Hz), beta (12 to 30 Hz) and gamma (>30 Hz) bands.

Linear classifiers tend to prefer multiple features. For the RC-based set-up on the other hand, one feature was selected for each dataset. This is due to the fact that forward feature selection algorithms tend to be sparse and that the first selected feature performed very well. Adding extra, less informative features distorts the states of the reservoir so that it is unable to achieve better performance. The feature that was selected for the GAERS data was the level 3 Daubechies 4 wavelet filter. For the PSE dataset, the energy in the beta band was selected. In Figure 4.6.B an example is given of the

wavelet filtered signal of the SWD shown in Figure 4.6.A.

### Rescaling

For each of the features independently, this signal is then subdivided in non-overlapping intervals, with length $L$, from which the foreground input signal $FG$ is calculated as the mean absolute value of these intervals. Since there is a high variability in signal amplitudes between different rats, a background signal is estimated for each of the EEG-features to serve as a reference level for rescaling. It is estimated from the foreground signal, for each past hour of EEG, as follows: $BG = \text{median}(FG)$. This is very similar to the background estimation of the Osorio-Frei algroithm (Osorio et al., 1998). It is a quantile-based estimation technique as proposed in Stahl et al. (2000), and is based on the assumption that epileptic seizures occur less than half of the time. As input for the classification algorithm the following equation is used:

$$I = \left\{ \frac{FG_f}{BG_f} \right\}, \forall f \in FS,$$

here $FS$ is the selected feature set. The rescaled and windowed foreground signal of an SWD is shown in Figure 4.6.C and is used as input for the classifier. For the GAERS data an interval with $L = 0.02$ s was determined during training, for the PSE dataset $L = 0.2$ s.

## 4.4.2 Classifier

The classifier consists of two parts: a reservoir that maps the input to a higher dimensional space and a linear readout.

### Reservoir

As mentioned in Chapter 2, each non-zero input sample to a reservoir excites this dynamical system and pushes the system to a new state. It maps the input data, which consists of only 1 feature vector, to a higher dimensional space. In this higher dimensional space the probability increases that the seizure and non-seizure samples are linearly

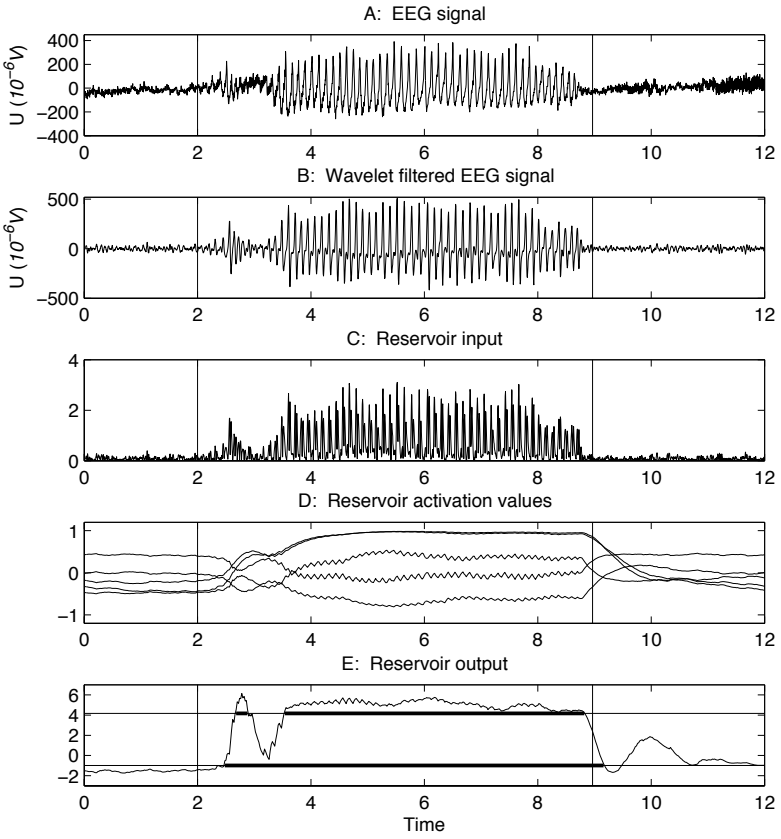


**Figure 4.6:** An example of a SWD caused by an absence seizure in GAERS. In (A) the EEG signal of one intracranial channel is shown. The seizure starts at Time = 2 s and stops at Time = 8.9 s. The preprocessing is shown in (B), the wavelet filtered signal, and (C): the rescaled absolute value of the filtered signal. (D) shows the activation values of 5 of the 200 reservoir neurons and (E) shows the generated output that was trained using BRR together with the two thresholds. Every sample above the high threshold is a detection, illustrated with the bold part of the high threshold line (the upper horizontal line). Samples neighboring these detected samples that are above the second threshold are used for marking and illustrated by the bold part of the low threshold line.

**Table 4.3:** The optimal reservoir parameters for both the GAERS and PSE data.

| Parameter | Value |
|---|---|
| spectral radius | 1.2 |
| input scaling | 0.3 |
| bias | 0.5 |
| leak rate | 0.05 |
| # neurons | 200 |

separable (Cover, 1965). For this set-up a reservoir of 200 neurons is used, this value was manually chosen and showed to give consistent performance during cross-validation on the training set.

There are several parameters that determine the dynamic properties of the reservoir. In Table 4.3 the optimal parameters for this task are shown. During optimization and as shown in Buteneers et al. (2010), these optimizations resulted the same optimal parameter values for the GAERS and the PSE data. Note that the bias and the spectral radius are very high. This shows that the optimal parameters push the reservoir into a highly non-linear regime. In Figure 4.6.D the activation values of 5 of the 200 neurons are shown when processing seizure data.

As discussed in Section 2.8.2, a reservoir has two main properties that contribute to its processing power. It has a quasi exponential decaying memory of its past inputs and it performs a non-linear transformation on these inputs. To determine whether it is a necessity to combine both properties, the reservoir is replaced with a linear time window (TW). This removes the non-linear transformation but keeps the memory property albeit without a decay. The width of the time window has been optimized on the training set.

## Readout

Reservoirs can be trained in different ways. To generate the output, a linear combination of the neuron activations in the reservoir is made,

resulting in the signal shown in Figure 4.6.E. To determine the best linear combination several algorithms can be applied. Usually ridge regression (RR) is used, but other forms of regularisation techniques can be used. Each regularisation technique mentioned in Chapter 3 has a specific purpose. To compare them for seizure detection they have all been tested on the dataset.

Class reweighted RR (CRRR) has been specifically designed to compensate for the imbalance in the dataset. As discussed in Chapter 2, RR can have difficulties when dealing with imbalanced datasets. CRRR reweights the influence of each class and thus results in a different hyperplane to separate the seizures from normal EEG.

Because a reservoir is a random set of interconnected neurons, it can occur that some neurons interfere with the readout, result in over-fitting on the noise they might generate and thus reduce the performance. To test this assumption, feature selection can be applied to select the optimal set of reservoir features or neurons. Because evaluating every combination of neurons would take months, two approximating algorithms are compared: RFFS and RBFS, respectively a regularized version of a forward and a backward feature selection algorithm.

In some situations it can occur that some seizure examples have properties that no other seizure example has. In that case it is undesired this seizure example influences the model in such a way that it becomes too much influenced by it, so that it performs worse on the more typical EEG. Bayesian relevance regression (BRR) iteratively scales the influence of each seizure example so that the model trained represents a seizure detection model that is dependant on the relevance of each of the examples.

### 4.4.3 Thresholds

To classify the continuous-valued output generated by the readout, two thresholds are applied: a high and a low threshold. Every sample above the high threshold is considered a seizure sample and thus part of a seizure (see Figure 4.6.E). Lowering the high threshold allows for a shorter detection delay at the cost of more false positives. To gain annotation precision a low threshold is used so that every sample

neighbouring a seizure sample that is above this low threshold, but below the high threshold, is also considered as part of a seizure.

To optimize these thresholds, the sample-based balanced error rate (BER) was used. This error rate is influenced equally by each class. This can be particularly useful if the dataset is imbalanced such as the EEG data used. It is the average of (1 - sensitivity) and (1 - specificity), where the sensitivity is the estimated probability that a positive is a true positive and the specificity is the estimated probability that a negative is a true negative. In mathematical notation this becomes:

$$\begin{aligned} Sens &= \frac{TP}{TP+FP} \\ Spec &= \frac{TN}{TN+FN} \\ BER &= \frac{(1-Sens)+(1-Spec)}{2} \\ &= \frac{1}{2}\left(\frac{FP}{TP+FP}+\frac{FN}{TN+FN}\right), \end{aligned}$$

where $TP$ is the number of true positive samples, $FP$ the number of false postive samples, $TN$ the number of true negative samples and $FN$ the number of false negative samples.

Because epilepsy data is very unbalanced, it contains significantly fewer positive than negative samples. The BER corrects for imbalance and has the effect that $FP \gg FN$. This might result in a higher number of FPPS than FNPS, but also in a lower detection delay. This is because the high threshold will be lower than when using error rates that do not correct for the imbalance in the data. An example of an error measure that does not correct for the imbalance is the zero-one loss (L01) which measures the percentage of misclassified samples:

$$L01 = \frac{FP+FN}{TP+FP+TN+FN}.$$

Both error measures will be evaluated to compare their effect on the performance.

### 4.4.4 Performance comparison

In order to evaluate the overall performance, the method was tested on the test set after training (on the training set). Although the experimental results below show the test performance, similar results and design decisions were achieved using cross-validation on the training set. Because of the random initialisation of RC, 10 different reservoirs were trained on each training set and the system that performed best on the training set was used for testing. Although Section 2.5 shows that the best train error does not necessarily result in the best test error, the training and test error are correlated for an optimally regularized system.

Table 4.4 shows that RC combined with its default training technique, RR, significantly outperforms both the AOFA and the method by Van Hese. The number of false positives and missed seizures is reduced by about a factor 10 and this with a very low variability between animals. Even the detection delay is significantly reduced to about 1 second for SWDs and 9 seconds for detecting the limbic seizures of the PSE rats. For the GAERS data 1 in 12 detections is a false positive and less than 7% of the seizures are missed. On the PSE data 1 in 5 detections is a false positive and only 0.3% of the seizures are missed.

The reservoir parameters from Table 4.3 show that the reservoir functions in a highly non-linear regime. This explains why, when a reservoir is replaced by a linear time window (TW), the performance is significantly worse. Note that, on average, the TW results are still better than what is achieved using the AOFA method. This shows that when linear regression is applied, and thus linear weights are trained on a TW, this more suited than only using the median of that TW, which is used by the AOFA.

As mentioned earlier, seizures are rare events. This means that there is more inter-ictal data in the training set than there is ictal data. Although CRRR is specifically designed to compensate for this imbalance, Table 4.4 shows that the imbalance is not necessarily a bad thing. All seizure detection systems from literature, even the detection systems for human data, are characterized by many false positives. Applying RR as opposed to CRRR makes sure that the

**Table 4.4:** The averages and standard deviations (between brackets) over the different rats of the FPPS, FNPS, detection delay and variance on the detection delay for all the methods

| **GAERS** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| RC-RR | 0.09 (0.16) | 0.065 (0.055) | 0.97 (0.33) |
| RC-CRRR | 0.15 (0.11) | 0.073 (0.068) | 0.49 (0.64) |
| RC-RFFS | 0.56 (0.79) | 0.023 (0.028) | 0.21 (0.77) |
| RC-RBFS | 0.069 (0.058) | 0.079 (0.068) | 0.71 (0.61) |
| RC-BRR | **0.065 (0.052)** | **0.071 (0.057)** | **0.79 (0.55)** |
| RC-BRR-L01 | 0.061 (0.051) | 0.079 (0.058) | 1.1 (0.4) |
| TW | 2.2 (3.0) | 0.020 (0.037) | 1.0 (1.1) |
| AOFA | 1.4 (3.0) | 0.18 (0.20) | 2.5 (0.6) |
| Van Hese | 3.5 (3.2) | 0.11 (0.07) | n/a |

| **PSE** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| RC-RR | 0.26 (0.21) | 0.003 (0.064) | 9.4 (3.4) |
| RC-CRRR | 1.0 (0.9) | 0.009 (0.018) | 7.9 (4.8) |
| RC-RFFS | 0.19 (0.24) | 0.11 (0.18) | 13 (5) |
| RC-RBFS | 0.32 (0.25) | 0.022 (0.053) | 9.4 (3.0) |
| RC-BRR | **0.13 (0.12)** | **0.005 (0.016)** | **9.4 (2.1)** |
| RC-BRR-L01 | 0.042 (0.52) | 0.016 (0.028) | 13 (2) |
| TW | 1.1 (1.1) | 0.026 (0.056) | 15 (4) |
| AOFA | 3.4 (3.8) | 0.032 (0.067) | 20 (4) |

importance of the non-seizure samples is not reduced in favour of the seizure samples. This has as a side effect that the model fits non-seizure better using RR than when CRRR is applied. Because there is a higher variability in non-seizure data, a better seizure detection model is found using RR.

Reducing the number of neurons that influence the output is shown to have a negative or almost no effect on the performance. The poor performance of RFFS could be explained by the fact that,

as many forward feature selection algorithms (Guyon and Elisseeff, 2003), it fails to find the constructive combinations of neurons since it only adds one neuron at a time. Backward algorithms on the other hand do not have this problem and should perform significantly better. Since this is not the case for the PSE dataset, it means that the difficulty lies elsewhere, namely in the error measure. Instead of optimizing the FPPS, FNPS and detection delay, the mean squared error is optimized. As shown in Section 2.3, the mean squared error is a somewhat bad indication for the error we try to optimize, therefore the results are according.

If the variability between examples in the training set is taken into account (RC-BRR) the performance improves. For the GAERS data, this technique achieves a detection delay of 0.8 seconds while detecting less than 1 in 16 false positives and missing only 7% of the seizures. On the PSE data, this method results in a detection delay of 9.4 seconds while about 1 in 8 detections is a false positive and only 0.5% of the seizures are missed. RC-BRR not only, on average, outperforms the other techniques, it also has a significantly lower standard deviation. This shows that the performance of this method is less influenced by differences between each animal in the dataset. It is thus better able to create a general model for seizure detection. The comparison with the RC-RR method shows that the common model created by BRR outperforms the method without BRR especially for the PSE dataset. This is because there is a high variability between the different animals. For the GAERS dataset there is however less variability between animals and thus the performance gain is limited. From these experiments it is thus clear that BRR is most suited to train the reservoir for seizure detection. Therefore it will be used throughout the rest of this work.

Optimizing the thresholds using the L01 as opposed to the BER clearly increases the detection delay and the number of FNPS and decreases the number of FPPS. This clearly shows the influence of the threshold on these error measures. The next section will dive deeper into the effects of changing the threshold.

**Table 4.5:** The fixed computation time, the optimization time and the number of removed features are given for the PSE dataset containing 23 animals. The computation time for RR and CRRR is grouped because they are equal. For the techniques based on RR, 60 regularization parameters were used. The results for the naive implementation of RBFS were not included because the computation time was more than 100 days.

| **PSE** | (CR)RR | | | RFFS | | | RBFS | | BRR |
|---|---|---|---|---|---|---|---|---|---|
| | naive | cov. | eig. | naive. | cov. | eig. | cov. | eig. | |
| $t_{fixed}$ (min) | | 0.3 | | | 0.3 | | 0.3 | | 0.3 |
| $t_{opt}$ (min) | 120 | 0.2 | 0.02 | 240 | 5 | 4 | 1700 | 6 | 0.7 |
| % rem. | 0 | 0 | 0 | 98 | 98 | 98 | 36 | 36 | 0 |

### 4.4.5 Computation time comparison

In Chapter 3 several computationally efficient algorithms were proposed for optimizing the regularisation parameter(s) and input features. In Table 4.5 the computational cost is shown for these different training techniques applied to the PSE dataset. It shows that the proposed eigendecomposition based algorithms clearly outperform the naive implementation and the covariance technique. Compared to the fixed computational cost for linear regression, 20 seconds, the optimization of the regularization parameter can be ignored. As shown in Buteneers et al. (2012a) this difference becomes even more prominent if more input features are used. Although FS does not improve the performance, Table 4.5 shows that the algorithm derived in Section 3.3.2 clearly outperforms the covariance method with respect to the computation time. The difference for the RFFS algorithm is less clear since only 2% of the features are selected. The computational cost of BRR is significantly higher than the cost of the eigenvalue based RR algorithm as is shown in Table 4.5. Since training the full system requires about 12 minutes, the added computational cost of the BRR algorithm is limited. The biggest contributors to computational cost during training are the reservoir simulation and the threshold opti-

mizations. Both require approximately 5 minutes.

## 4.5 Reducing the detection delay

For some tasks it is paramount to have a low detection delay. If for example a seizure detection method is used to trigger anti-epileptic treatment it is crucial that the detection delay is very low for the treatment to be efficient (Hammond et al., 1992).

The RC based and AOFA methods can be altered to achieve a lower detection delay at the expense of a higher number of FPPS and coincidentally a lower number of FNPS. Each method has one specific parameter that has the most influence. For the RC based methods this parameter is the high threshold. The main cause of the detection delay of the AOFA method is the parameter $T_s$ which indicates the minimal duration of a seizure. However, if this parameter is decreased manually to reduce the detection delay, the threshold needs to be re-optimized. Figure 4.7 shows the results of these alterations for the RC-BRR method and the AOFA method. In the top graphs of Figure 4.7 the FPPS is plotted as a function of the detection delay for the GAERS and PSE dataset respectively. The bottom graphs of Figure 4.7 show the FNPS as a function of the delay. Settings that resulted in more than 4 FPPS were not included.

If the algorithms are altered to result in a lower detection delay, this comes at the cost of more false detections as shown in Figure 4.7 (top graphs). This is because the first few samples of ictal EEG better resemble normal EEG. The number of missed seizures on the other hand decreases because less distinguishable seizures will also be detected. Figure 4.7 also shows that, for a given number of FPPS or FNPS, RC-BRR is able to detect the seizures with less delay than the AOFA method. If 4 FPPS are allowed, i.e. 1 in 5 detections is a true positive, RC-BRR achieves an average detection delay of 0.2 s and 2.1 s and the AOFA-method will take more than 2 s and 15 s to detect a seizure on the GAERS and PSE data respectively. This reduction in detection delay significantly reduces the number of missed seizures.

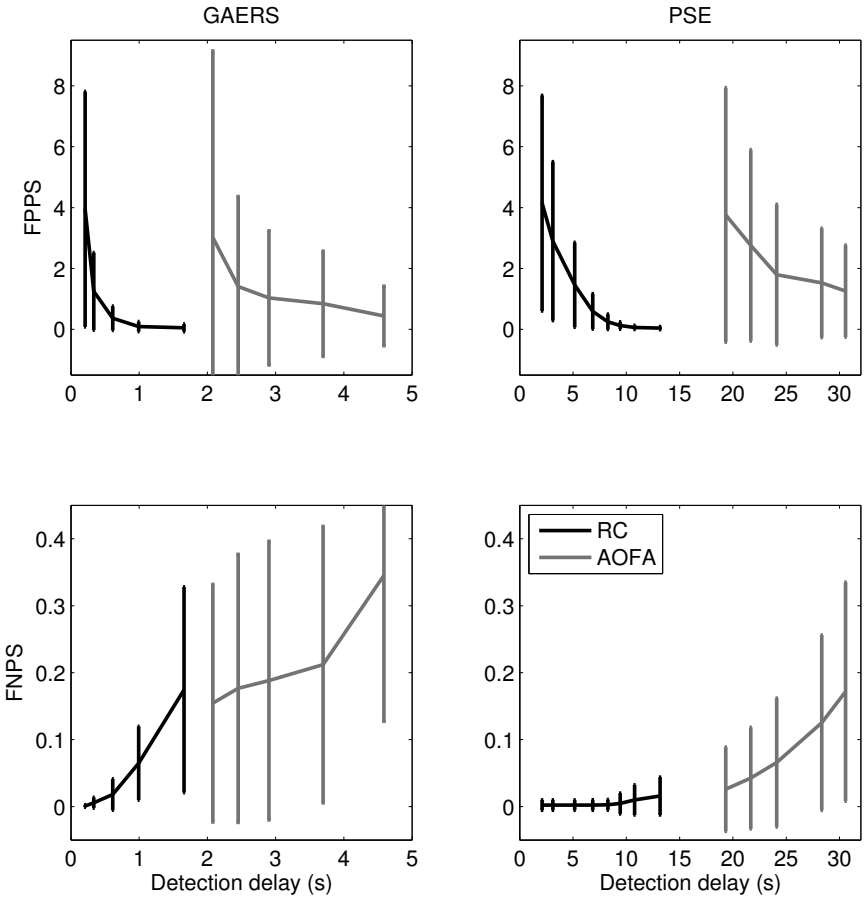


**Figure 4.7:** On the top the number of False Positives Per Seizure and on the bottom the number of False Negatives Per Seizure as a function of the detection delay in seconds for the GAERS dataset on the left and the PSE dataset on the right.

On the GAERS dataset only 1 in 1000 seizures were missed. On the PSE dataset the number of missed seizures was reduced to 1 in 500. When minimal FPPS and FNPS are required we see that RC-BRR is the only method that is able to achieve less than 10% FPPS and FNPS simultaneously.

Adapting the methods to achieve a lower detection delay deteriorates the annotation accuracy of the methods. One could use one

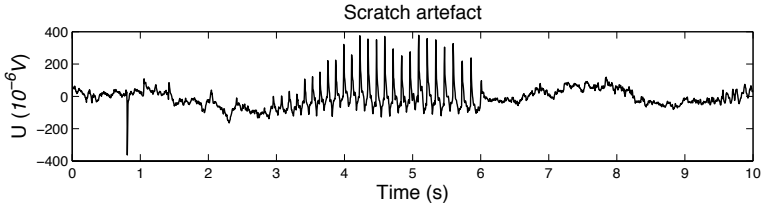


**Figure 4.8:** An example of a scratch artefact which starts at time = 3s and ends at time = 6s.

set-up for real-time detection and another for annotation. Since for the RC-BRR method no new training is required and only the high threshold is altered to achieve a lower detection delay, the same set-up can be used for both tasks. One only needs to use two different high thresholds: one for real-time detection and one for annotation.

If for marking purposes, a very low FPPS and FNPS is required, another approach can be used to limit the input from the user. Let us assume that in this situation it is acceptable to miss 1 in 500 seizures. If the RC-BRR method is used, this results in a system that detects 2.5 and 4 FPPS for the GAERS and PSE datasets respectively. An experienced encephalographer can now go over these detections and remove the false detections. This means that the amount of data that needs to be reviewed for each dataset is about 3.5 and 5 times the length of a seizure, respectively. For the GAERS dataset this results in 75% less data that needs to be reviewed and for the PSE dataset this is a reduction of 90%. For large datasets, only a small fraction of the remaining time is required to build the training set.

## 4.6 The 'golden standard'

It is a well known fact that humans often disagree when marking epileptic seizures. We took a small dataset used to train students in marking epileptic seizures of 3 PSE rats. It contains 24 hours for each rat with in total 72 seizures and 183 artefacts. Most of these artefacts are scratch artefacts as the one shown in Figure 4.8. As golden standard we asked 4 experienced encephalographers to mark the data and come to an agreement. In table 4.6 we show the performance of

**Table 4.6:** A comparison in performance of the RC-BRR method and two experienced encephalographers (E1 and E2) on a small student dataset.

| $\mathbf{PSE}_{student}$ | FPPS | FNPS |
|---|---|---|
| RC-BRR | 0.15 (0.17) | 0.09 (0.14) |
| E1 | 0.01 (0.02) | 0.029 (0.029) |
| E2 | 0.13 (0.18) | 0.010 (0.017) |

our method, trained on the training sets of dataset C, D and E, and the results of 2 experienced encephalographers, who had never seen the data before, against this golden standard.

Table 4.6 reaffirms that researchers often disagree when marking EEG. Even though RC-BRR performed worse than both experienced encephalographers, it is fair to say that the performance of our method is comparable with at least one of the encephalographers. The artefacts that caused false positives were all scratch artefacts and often coincided with the errors made by the encephalographers. Other artefacts or sleep spindels, rhythmic activity that occurs during sleep, did not generate any false positives.

To train the system and process the roughly 2500 hours of EEG it takes a 2.6 GHz Core 2 Quad machine (with 8 GB RAM) 7 hours of running slightly optimized Python code. A well trained experienced encephalographer is at least 11 times slower, being able to process about 4 hours of EEG from 8 rats simultaneously in one hour.

## 4.7 Stimulation artefacts

Epileptic seizure detection methods are often used to trigger neurostimulation such as deep brain stimulation or vagus nerve stimulation. These stimulation paradigms can cause stimulation artefacts on the EEG as shown in Figure 4.3. The EEG of several animals from study C contains periods where stimulation is applied. These animals are grouped in the subset C*. In Table 4.7 the results are shown for the

**Table 4.7:** A comparison in performance of the RC-BRR method on EEG with or without stimulation artefacts. It was tested on the animals from study C that were stimulated: C*$_{clean}$ represents the EEG data without stimulation artefacts and C*$_{stim}$ the EEG data with stimulation artefacts.

| **PSE** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| C*$_{clean}$ | 0.084 (0.080) | 0.013 (0.027) | 7.1 (2.5) |
| C*$_{stim}$ | 0.048 (0.069) | 0.043 (0.043) | 6.7 (3.3) |

part of the C* subset without stimulation artefacts C*$_{clean}$ and the part with simulation artefacts C*$_{stim}$.

Counter intuitively, stimulation artefacts do not increase the number of FPPS as shown in Table 4.7. This is possibly due to the fact that a stimulation of 130 Hz was applied which does not fall within the beta band that was used for pre-processing. The artefacts however do increase the background signal level which results in a higher number of FNPS, a lower sensitivity and thus also less FPPS. However, the increased background scaling has no influence on the detection delay. If the exact time and duration of the stimulation is known, one could ignore these samples while estimating the background signal.

## 4.8 Depth versus epidural EEG

Datasets A, C, D and E were recorded using depth electrodes, dataset B on the other hand was recorded using epidural electrodes. To compare whether the system performs better when the EEG is recorded using depth or epidural electrodes we present the results for the RC-BRR method on datasets A and B separately in Table 4.8.

From Table 4.8 one can infer that there is a difference in performance when depth electrodes are used as opposed to epidural electrodes. The system achieves a slightly lower detection delay with a significantly lower number of FPPS and FNPS. In our experience this

**Table 4.8:** A comparison in performance of the RC-BRR method on depth and epidural EEG from datasets A and B respectively.

| **GAERS** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| A | 0.039 (0.026) | 0.024 (0.021) | 0.82 (0.22) |
| B | 0.15 (0.23) | 0.11 (0.05) | 1.1 (0.4) |

is due to the fact that there is less noise in the EEG which makes that the difference between ictal and inter-ictal EEG is more profound in terms of signal strength and signal shape. For optimal performance it is therefore advised to use the RC-based technique in combination with depth electrodes.

## 4.9 Active learning

Active learning (AL) is most commonly used to allow an algorithm to be trained without requiring a lot of annotated data. It is usually trained on a small training set and afterwards the system is used to analyse the unmarked data. If the system is uncertain about its predicted output, it will stop reviewing the output and ask for input from the user. The user can now give the exact output so that this data can be used to extend the training set. After retraining, the rest of the data is evaluated and the process repeats itself until there are no uncertain or unmarked data points left. This approach dramatically reduces the required training set size and can achieve similar performance (Balakrishnan and Syed, 2012).

Another approach to AL that can be used is to only learn from mistakes and will be used in this work. To find these mistakes however, all the data needs to be reviewed. Because a seizure is a rare event and there are not many FPPS, it is more appealing to only review the positive outputs generated by the system as suggested in Section 4.5. Once a false positive is identified, it can be added to the training set. Since the true positives are reviewed anyway and in order to keep

a form of balance between the positive and negative examples in the training set the positive examples can be added as well. This allows for the system to not only better learn the properties of the inter-ictal EEG but also to better learn which properties define a seizure.

Figure 4.9 shows the performance of the RC-BRR system trained on the full dataset and on randomly selected subsets of the training set. For the GAERS training data, which contains 23 examples, 2, 5 and 10 examples were randomly selected. For the PSE dataset, containing 220 examples, 3, 10 and 30 examples were randomly selected. The experiments were repeated 5 times to achieve statistically relevant results. Figure 4.9 shows that with a smaller training set, there are more FPPS and FNPS. For the detection delay there is however no general rule that holds for both datasets. When on the same small training sets AL is applied (marked with AL in Figure 4.9), it is clear that AL improves performance. Although using the full training set is still the best option, using AL comparable performance is achieved, even for a very small dataset, while reducing the time needed to build a training set significantly. Because only positives are evaluated, the number of FPPS is less or equal to when the full training set is used. The number of missed seizures is increased, but is still significantly lower than when no AL is applied. As discussed in Section 4.5, the number of missed seizures can be further reduced by lowering the high threshold, such that more detections occur and less seizures are missed. This introduces only a limited amount of extra work by the encephalographer.

Even though the RC-BRR method shows comparable performance to at least one of the encephalographers from Section 4.6, it might happen that after training on the full training set, it is still required to review the data marked by the seizure detection system as suggested in Section 4.5, to remove the falsely detected and missed seizures. Because there is more inter-ictal EEG than ictal EEG that needs to be reviewed, it requires more work to find the missed seizures than to remove the false positives. As shown in Table 4.4 this seems most suited for the PSE dataset where the RC-BRR method has significantly more FPPS than FNPS. Once a detection is reviewed, whether it is a correctly detected seizure or a false positive, it can again be added to the training set to retrain the system. This will hopefully

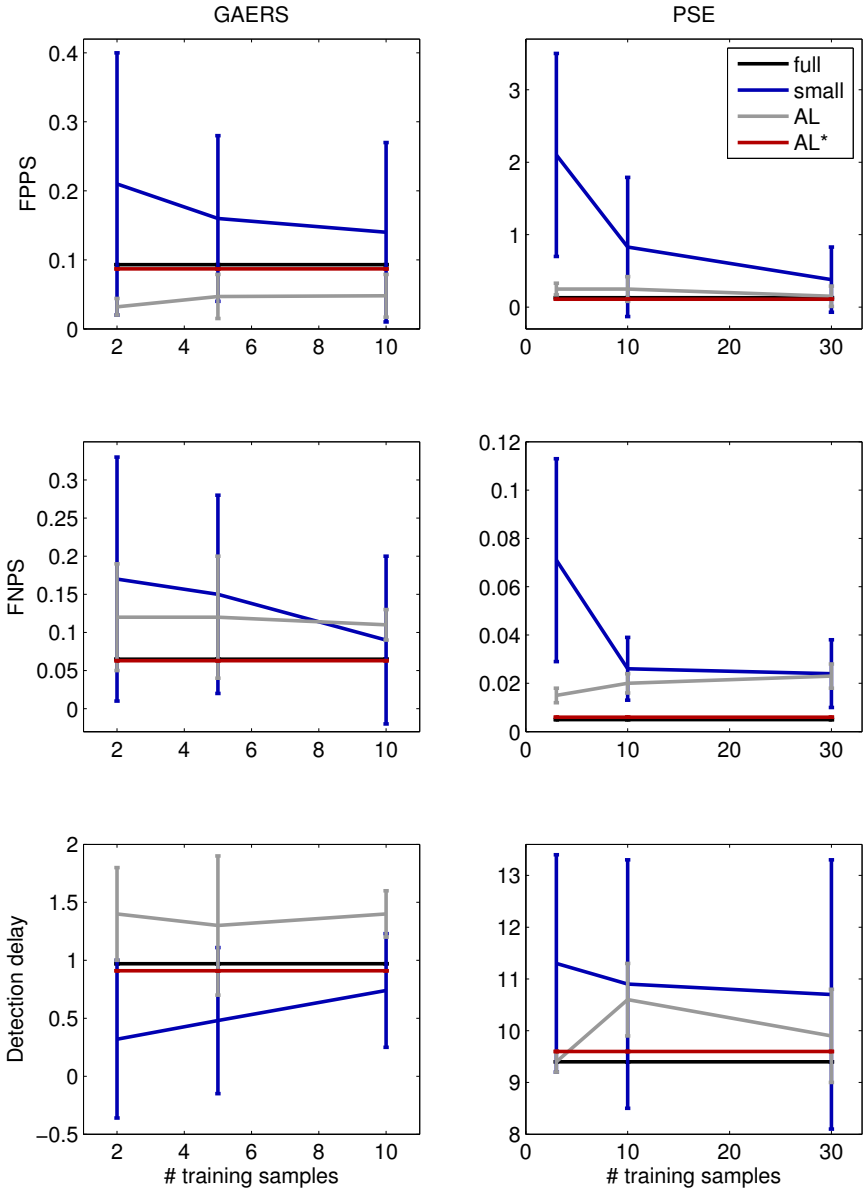


**Figure 4.9:** The results for the system trained on the full dataset with (AL*) or without active learning on the test set, or trained on a small dataset, with no further training or active learning (AL). The variance is computed over 5 random selections of training set samples.

allow the system to better learn inter-ictal data as well as ictal data.

To evaluate the effect of this type of AL (AL* in Figure 4.9) we set up the experiment as follows. The system is first trained on the training set of all animals. Next the system is evaluated on the first hour of the test set. The results are used to evaluate the performance and to determine the false positives and true positives. The false and true positive EEG parts are then added to the training set with their correct outputs and the system is retrained. This is repeated for every hour of EEG data from this animal. For the other animals the same procedure is repeated starting from the default RC-BRR system trained on the training set without the previously added samples. Figure 4.9 compares the performance of RC-BRR with or without AL on the test set. Since both are barely distinguishable on the figure, it shows that the effect of AL on a fully trained system is minimal. This suggests that using a training set as described in Section 4.1, allows the system to reach its optimal performance.

## 4.10 Conclusion

From this chapter we can conclude that RC outperforms state-of-the-art detection techniques on the iEEG of animal models. It has significantly less false positives and false negatives as well as a lower detection delay. In the default configuration, 1 out of 12 and 1 out of 9 detections is a false positive and and misses 1 in 13 and 1 in 200 seizures are missed for GAERS and PSE rats, respectively. It achieves an average detection delay below 1 second in GAERS and less than 10 seconds in the PSE data. This detection delay and the number of missed seizures can be further decreased when a higher false positive rate is allowed. If 4 FPPS are allowed, i.e., 1 in 5 detections is a true positive, the detection delay is reduced to 0.2 and 2 seconds, respectively, and only 1 in 1000 and 1 in 500 seizures is missed. Although this set-up has many FPPS it can also be applied to mark epileptic seizures with a very high accuracy. Since almost no seizures are missed, a user only needs to remove the false positives. This leads to a reduction of, 75 and 90% of data that needs to be

reviewed.

EEG recordings are known to contain many artefacts which usually result in many false positives. The presented technique has been shown to be robust against signal artefacts and high frequency stimulation artefacts. However, better performance is still achieved on data without artefacts. The performance is also dependent on the way the EEG has been recorded. Depth EEG electrodes show a better yield than epidural electrodes.

Although its performance is somewhat comparable to that of encephalographers, the technique is still outperformed by its human counter parts. However, it avoids the time-consuming manual review and annotation of EEG and can be incorporated in a closed-loop treatment strategy. It is therefore suited for automatic seizure detection based on iEEG and may serve as a useful tool for epilepsy researchers.

For the system to achieve the best possible performance it is advised to use a training set with about 10 seizures per animal and about 10 times more inter-ictal EEG as opposed to ictal EEG. However, building this training set requires manual annotation by an experienced encephalographer. Using active learning similar performance can be achieved while requiring only a fraction of the work. It starts with a training set of about one tenth the size, after which only the rare false and true positives need to be evaluated, until about 10 seizures per animal have been annotated by the encephalographer.

# 5 Seizure detection in human EEG data

The previous chapter has shown that state-of-the-art performance can be achieved with RC on iEEG data from animal models. This chapter will apply the acquired knowledge to human data.

As mentioned in the first chapter, there have been many attempts at building seizure detectors for humans. In literature, the distinction is made between a patient specific seizure detector that is optimized for one patient in particular and a patient unspecific or general seizure detector that is supposed to work for all epilepsy patients. Since RC has achieved promising results in the previous chapter, we now investigate whether state-of-the-art performance can be reached on human data for both types of seizure detectors.

## 5.1 Materials

In this chapter two public datasets are used: the CHB-MIT Scalp EEG Database (Goldberger et al., 2000) and the iEEG Database recorded for the Seizure Prediction Project Freiburg. Although both datasets contain EEG artefacts, no files were removed and the artefacts are included as inter-ictal EEG.

### 5.1.1 CHB-MIT Scalp EEG Database

This dataset, further referred to as the Scalp dataset (Goldberger et al., 2000), was collected at the Children's Hospital Boston. It consists of EEG recordings from 23 paediatric patients with intractable seizures. They were monitored for up to several days following withdrawal of anti-seizure medication in order to characterize their seizures and assess their candidacy for surgical intervention.

All signals were sampled at 256 Hz with 16-bit resolution. Most files contain 23 EEG signals (24 or 26 in a few cases) and the International 10-20 system of EEG electrode positions and nomenclature was used for these recordings. The recordings are grouped into 24 cases. They were collected from 23 patients: 5 males, ages 3-22 and 17 females, ages 1.5-19. Case 21 was obtained 1.5 years after case 1 from the same female patient.

Although the data was recorded consecutively, the recorded EEG channels are not always the same. To create a homogeneous dataset for each patient, additional channels or files with an incompatible montage were left out of the dataset. This resulted in a dataset with 18 similarly positioned EEG channels per patient[1]. In Table 5.1 the remaining total length of the data, the number of seizures and their average length are given for each case.

### 5.1.2 iEEG Database Freiburg

The iEEG database[2] was recorded for the Seizure Prediction project at the Epilepsy Center of the University Hospital of Freiburg, Germany. It contains invasive EEG recordings of 21 patients with medically intractable focal epilepsy: 13 females, ages 10 to 50 and 8 males, ages 14 to 47. The data were recorded during an invasive pre-surgical epilepsy monitoring. In eleven patients, the epileptic focus was located in neocortical brain structures, in eight patients in the hippocampus,

[1] These channels are the ones shown in Figure 1.6 together with the channels *Fz-Cz* and *Cz-Pz*.

[2] Available at https://epilepsy.uni-freiburg.de/freiburg-seizure-prediction-project/eeg-database.

**Table 5.1:** The number of hours of EEG data, seizures and their average length (in seconds) for each case in the Scalp dataset.

| Case | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| hours | 42 | 36 | 38 | 157 | 39 | 67 | 66 | 20 | 69 | 50 | 35 | 21 |
| seizures | 7 | 3 | 7 | 4 | 5 | 10 | 3 | 5 | 4 | 7 | 3 | 27 |
| length | 64 | 58 | 58 | 93 | 113 | 16 | 109 | 185 | 70 | 65 | 270 | 38 |

| Case | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| hours | 33 | 26 | 40 | 19 | 21 | 36 | 30 | 29 | 33 | 31 | 26 | 22 |
| seizures | 12 | 8 | 20 | 10 | 3 | 6 | 3 | 8 | 4 | 3 | 7 | 16 |
| length | 46 | 22 | 101 | 9 | 99 | 54 | 80 | 38 | 51 | 69 | 62 | 33 |

and in two patients in both. In order to obtain a high signal-to-noise ratio, fewer artefacts, and to record directly from focal areas, intracranial grid-, strip-, and depth-electrodes were used. For each patient, the recordings of three focal and three extra-focal electrode contacts are available.

## 5.2 Evaluation measures

The gold standard used to compare the different detection methods is the scoring provided with each dataset. As a measure for the number of seizures that were falsely detected, the number of false positives per seizure (FPPS) is measured. The number of missed seizures are measured in false negatives per seizure FNPS. The detection delay is only determined for correct seizure detections. Detections that occur before a seizure, but where the detection is interrupted before the beginning of the seizure, are considered as false positives. Since the data is subdivided into 1 hour files, either the time since the first inter-ictal sample after the previous seizure is used as a lower bound, or the time since the first sample of the file, if there is no seizure preceding the detected seizure in the current file. As an upper bound, the last

**Table 5.2:** The number of hours of EEG data, seizures and their average length for each case in the iEEG dataset.

| Patient | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| hours | 31 | 30 | 33 | 34 | 34 | 32 | 31 | 28 | 36 | 36 | 33 |
| seizures | 4 | 3 | 5 | 5 | 5 | 3 | 3 | 2 | 5 | 5 | 4 |
| length | 13 | 118 | 93 | 87 | 45 | 67 | 154 | 164 | 113 | 411 | 157 |

| Patient | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 |
|---|---|---|---|---|---|---|---|---|---|---|
| hours | 58 | 28 | 32 | 34 | 36 | 40 | 39 | 37 | 38 | 37 |
| seizures | 4 | 2 | 4 | 4 | 5 | 5 | 5 | 4 | 5 | 5 |
| length | 55 | 158 | 216 | 145 | 121 | 86 | 14 | 13 | 86 | 83 |

marked sample of the to be detected seizure is used. The detection delays in this work represent the actual delay that would be achieved during on-line testing and thus includes the time required to gather the data used for the windowed energy features. Since all tested methods are computationally inexpensive to evaluate, the computation time is of the order of milliseconds and can be ignored.

To optimally exploit the relatively short datasets used in this work, these measures are computed using leave-one-hour-out cross-validation. Each of the evaluation measures is computed for each patient individually. The average and standard deviation is computed over all the patents where each patient has the same influence on that result, independent of the amount of data that was recorded for this person. In the figures, the average is represented by A. In the tables the standard deviation is given between rounded brackets.

## 5.3 Methods from literature

In Chapter 1, a broad range of seizure detection methods has been described. In this chapter, 3 of these methods will be compared with RC: the Osorio-Frei algorithm (OFA) (Osorio et al., 1998), the Reveal algorithm (Wilson et al., 2004) and the method by Shoeb et al.

(Shoeb, 2009).

### 5.3.1 Osorio-Frei

As mentioned in the introduction, the method presented in Osorio et al. (1998) is the most frequently cited seizure detection method. It thanks its popularity to both its simplicity and its effectiveness. It was specifically designed to detect seizures on the iEEG. Therefore it will be applied on the iEEG dataset. A technical description of the algorithm is given in Section 4.3.1. In Osorio et al. (1998) it was shown to be able to detect all the seizures, without detecting any false positives, with an average detection delay of 2.1 s. Since these results were achieved by fine tuning the threshold and minimal duration on the test set, these results might not be a good indication for the performance on the iEEG dataset used in this work. However, in Osorio et al. (2002) similar detection delay while missing none of the seizures and only a few false positives was achieved on a different dataset containing 14 patients. On the iEEG used in this work it was able to detect 76% of the seizures with an average detection delay of 21 seconds while detecting 32 FPPS.

### 5.3.2 Reveal

The Reveal algorithm by Wilson et al. (2004) is a non patient specific seizure detection algorithm. It is included in the comparison because it has been evaluated in Shoeb (2009) on the scalp dataset used in this work. Only a brief description is given here. For more details we refer to literature.

The method uses the matching pursuit algorithm which converts the EEG into the sum of overlapping 'atoms'. They are each localized in time and in frequency and can be thought of as a sparse time frequency decomposition of the EEG. These atoms are evaluated using 6 manually selected rules that determine whether a certain signal is part of a seizure or not. For 4 of these rules the parameters are trained using neural networks. In Wilson et al. (2004) it is shown to be able to detect 76% of the seizures. It detected only 2.6 false positives

per 24 hours on a dataset of non-epileptic patients. Since epilepsy patients have more rhythmic, non-seizure activity, these results are not representative for the number of false positives detected on the EEG of epilepsy patients.

The authors seem to indicate that the test set is used to train the neural networks. The Reveal algorithm has been tested on the first 23 cases of the Scalp dataset in Shoeb (2009). There it missed 35% of the seizures which is the same order of magnitude as the 24% missed seizures as published in Wilson et al. (2004). It had however 18 false positives per seizure in the least sensitive setting which is about 10 times more than was suggested in Wilson et al. (2004). Similar findings on the same dataset were reported in Balakrishnan and Syed (2012).

### 5.3.3 Shoeb et al.

In Shoeb (2009) a patient specific seizure onset detection algorithm was presented that can be considered the current state-of-the-art for patient specific seizure detection.

The EEG is first preprocessed by applying a frequency filter bank. It contains 8 frequency filters of 3 Hz wide and ranges from 0.5 to 24.5 Hz. After applying these filters, the average energy is calculated over windows of 2 seconds wide with an overlap of 1 second. Before these energy features are used as input for a support vector machine (SVM), the logarithm is taken and the signal is rescaled to have zero mean and unit variance[3]. The logarithm ensures that the data is more or less Guassian distributed. As input for the SVM, 3 consecutive windows are used (without overlap). For training, only the first 20 s of each seizure are used, the rest of the seizure data is ignored. Although the same hyper-parameters were used for all the patients, Shoeb (2009) does not mention how these parameters were selected.

In the original publication, the method achieved about 0.5 false detections per seizure and detected 91% of the seizures with an average

[3]This is not indicated in Shoeb (2009), but it was communicated by the authors of Balakrishnan and Syed (2012). Not implementing this step results in a system that is unable to detect seizures.

detection delay of 4.6 s on the Scalp dataset used in this work. There is also no indication in the work that the time to calculate the windowed energy features is taken into account. This would cause the detection delay to increase with 1 second.

An attempt at reconstructing these results on a subset of the Scalp data was done in Balakrishnan and Syed (2012). Here, only 3% of the seizures were missed, 1.4 false detections per seizure were detected and a detection delay of 7.9 s was achieved. Using our own implementation of this method on the full Scalp dataset resulted in a system that missed 28% of the seizures, had an average detection delay of 16 seconds and detected 0.28 FPPS if the full seizure was used for training. If only the first 20 seconds were used, 45% of the seizures were missed with 0.11 FPPS and an average detection delay of 13 seconds. The reason for these poor results is unclear. Possibly the slightly different dataset used in this work lies at the origin. Here, only 18 channels per patient were used and no files were removed apart from the files with incompatible EEG channels. The dataset used in Shoeb (2009) contains fewer seizures, which indicates that some files were removed. Whether these files contained artefacts or other abnormalities is unclear. Another possible reason for these significantly different results is that essential technical details are missing in Shoeb (2009). As noted earlier, taking the logarithm before processing the data is paramount for the system to function and was not mentioned in Shoeb (2009). Possibly other details are also missing.

## 5.4 Set-up of the proposed method

The set-up used for the human data is very similar to the one used for animal models. The data is first pre-processed, followed by a reservoir and a readout. Finally, the output is passed through the same thresholding technique.

### 5.4.1 Preprocessing

To preprocess the data, the same preprocessing method is used as the method described in Section 5.3.3. The signal is filtered using a filter bank with 3 Hz wide filters. For the Scalp dataset, the same 8 filters are used, ranging from 0.5 to 24.5 Hz. For the iEEG dataset 16 filters are used, ranging from 0.5 to 48.5 Hz. This broader range was selected because there is no frequency attenuation by the skull on iEEG data (Grewal and Gotman, 2005). This is for instance useful for seizures with a main frequency above 25 Hz such as the one shown in Figure 1.10.

After processing the data using a filter bank, the average energy is, similarly to Shoeb (2009), computed for windows of 2 seconds wide with an overlap of 1 second. Again the logarithm of this data is taken, after which the data is normalized. The windowing adds an extra delay of 1 second. As the state of a reservoir contains information about past inputs, only one window at a time is used as input for the reservoir. This is opposed to the 3 consecutive time windows used as input for the SVM based method by Shoeb (2009).

Although other features and feature selection could possibly improve results, it has several advantages to use the features discussed above:

- It is easier to compare both ML techniques with regards to their performance if they use the same features.
- Since feature selection is a time consuming task, avoiding it reduces the amount of training time.
- If the same features are used for all patients, a model trained on one patient is easily mapped onto another patient.
- These features have the advantage that they are easily implemented on hardware using a few basic analogue resistors, capacitors and transistors.

**Table 5.3:** The optimal reservoir parameters for the human data.

| Parameter | Value |
|---|---|
| spectral radius | 0.4 |
| input scaling | 0.02 |
| bias | 1 |
| leak rate | 0.2 |
| # neurons | 1000 |

### 5.4.2 The reservoir

Similarly to the system in the previous chapter, a reservoir is used to map the input features to a higher dimensional space. The reservoir parameters used are given in Table 5.3. These parameters were optimized on a scalp EEG dataset containing 24 hours of data and 8 seizures from a patient under pre-surgical evaluation at the Ghent University Hospital. Selecting optimal reservoir parameters on each training set, would be more logical, but for one experiment there are 1721 training sets. Optimizing the reservoir parameters for each of them is computationally not feasible and therefore a patient, not belonging to the Scalp and iEEG datasets, was used.

As shown in Section 2.8.2, the bias parameter used, indicates that the reservoir functions in the non-linear area of the hyperbolic tangent function. The spectral radius and leak rate suggest that the task requires a short memory of the past input. This can be explained because only one sample is used per second, which is constructed using a 2 second wide window.

### 5.4.3 Readout and threshold

For these stages the same setup as described in the previous chapter is used. The reservoir readout is trained using Bayesian relevance regression (BRR) which is described in more detail in Section 3.4. The output is post-processed with the dual threshold technique described

in Section 4.4.3.

## 5.5 Patient specific model

In a first set of experiments we compare the patient specific seizure detection method presented in Shoeb (2009) with the method presented in this chapter. To determine the performance of the patient specific model, leave-one-hour-out cross-validation is used. For every one-hour sample of EEG data, the system is trained on all other data samples and tested on this sample. For each method, two set-ups are compared: the normal set-up where all seizure samples are included in the training set, and the onset set-up (marked with O), where only the first 20 seconds of a seizure are used for training. The rest of the seizure is ignored for training, but not for testing.

The results of these experiments for the Scalp and iEEG dataset are shown in Figures 5.1 and 5.2, respectively and in Table 5.4. From these figures one can conclude that the performance of the presented method is characterized by more FPPS, while achieving significantly fewer FNPS and a slightly lower detection delay. The optimal method cannot be selected since there is no clear winner in these figures. One general rule that can be extracted from these figures is that using only the onset of a seizure for training reduces the detection delay.

The RC based method was unable to detect any of the seizures in only 2 patients from the iEEG database. These were the only 2 patients for which there were less than 3 seizures recorded. This means that the system needs at least 2 seizures in the training set. The SVM based method on the other hand failed to detect any seizures in 10 patients, 4 patients from the Scalp dataset and 6 patients from the iEEG dataset. The SVM onset method was even unable to detect seizures in 19 patients or in more than 40% of the patients.

As shown in the previous chapter, increasing the high threshold of the RC based method decreases the number of FPPS and increases the number of FNPS and the detection delay. Lowering the high threshold has the opposite effect. Figure 5.3 shows that when the high threshold is altered a broad range of properties can be achieved

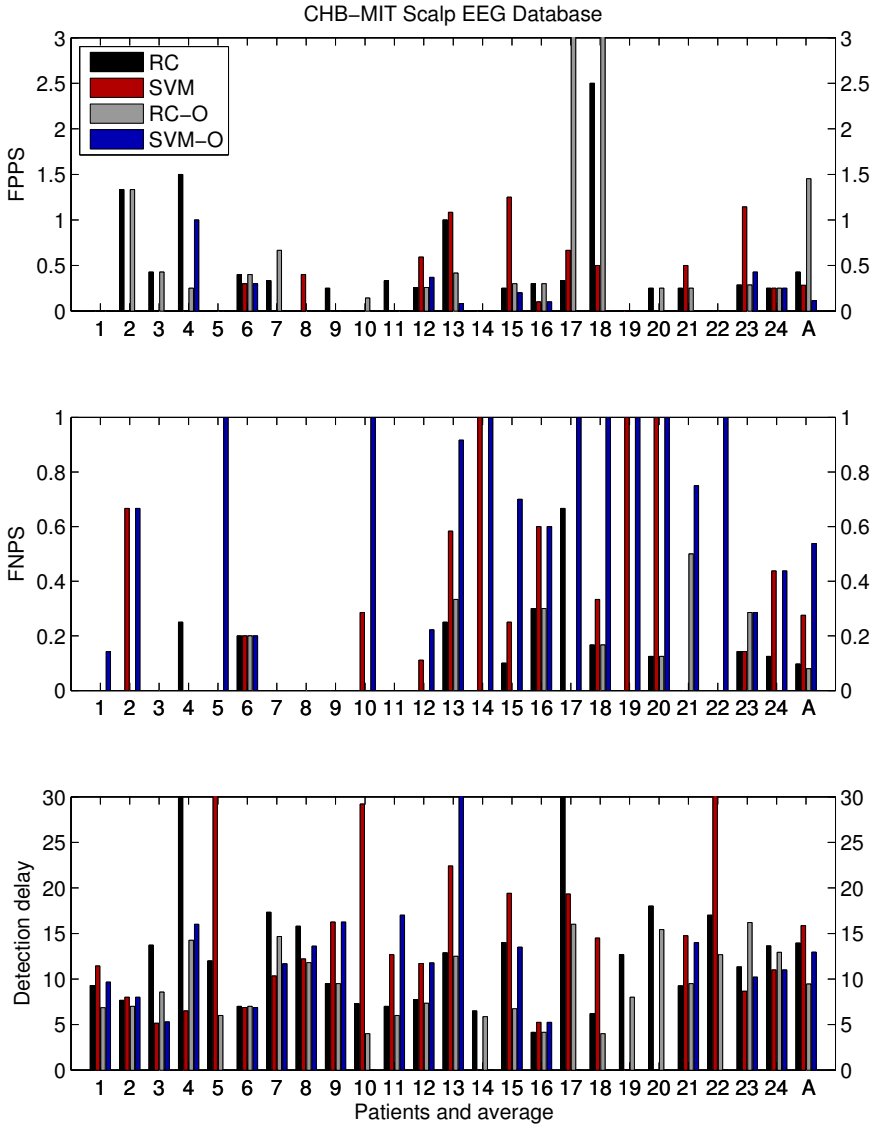


**Figure 5.1:** The FPPS, FNPS and detection delay for the patient specific seizure detection techniques on the Scalp dataset. The techniques marked with 'O' are only trained on the seizure onset. The number of FPPS shown is limited to 3, and the detection delay is cut off at 30 seconds.

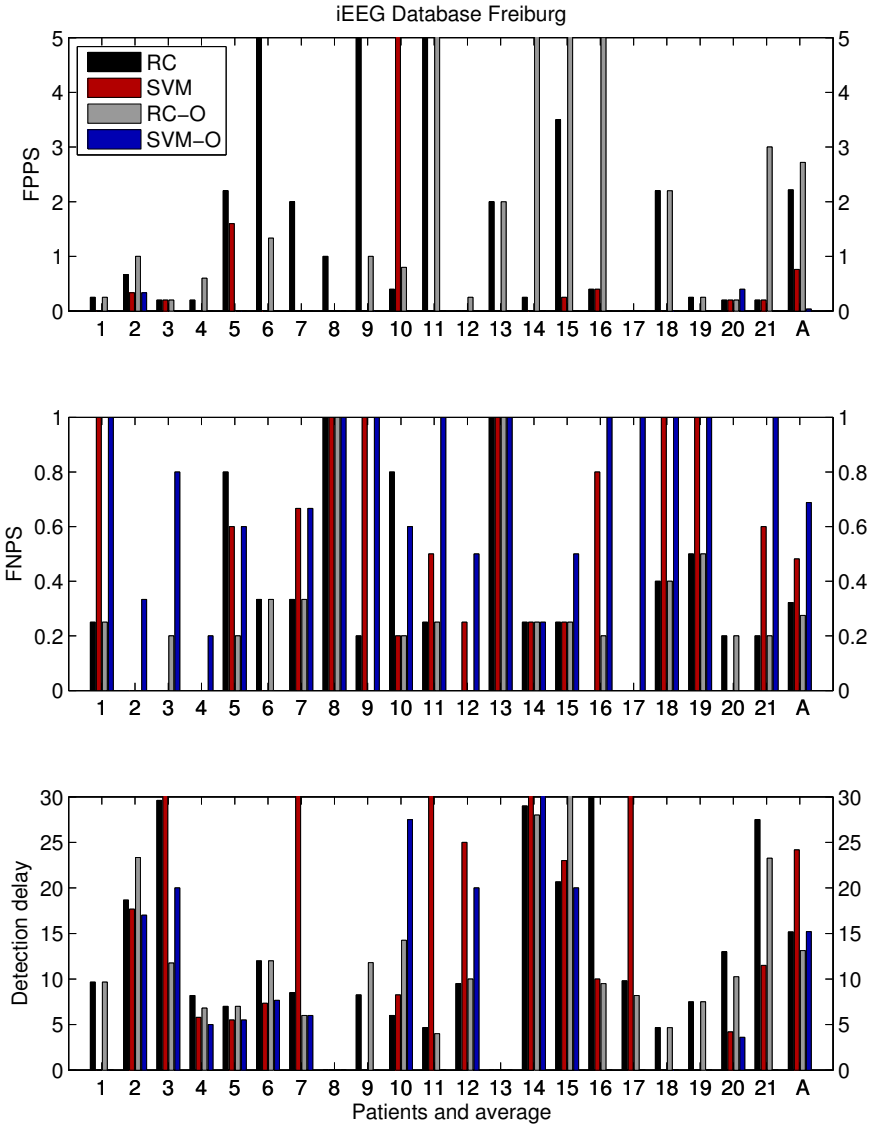


**Figure 5.2:** The FPPS, FNPS and detection delay for the patient specific seizure detection techniques on the iEEG dataset. The techniques marked with 'O' are only trained on the seizure onset. The number of FPPS shown is cut off at 5 and the detection delay is cut off at 30 seconds.

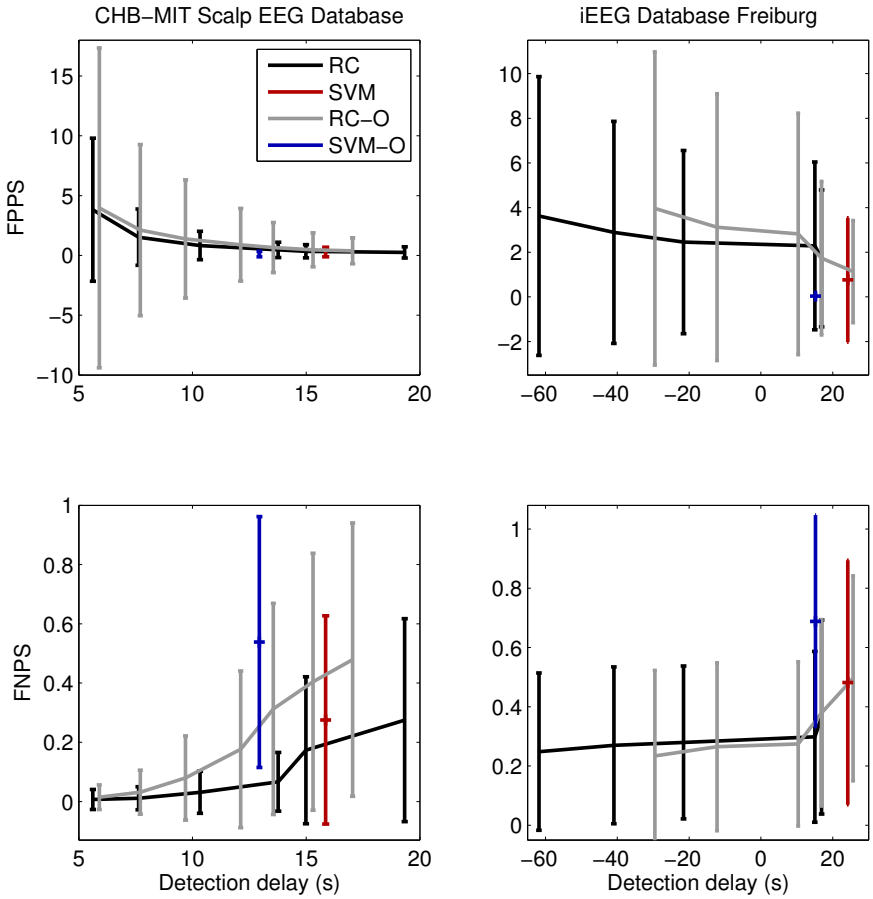


**Figure 5.3:** The FPPS, FNPS versus the detection delay for the patient specific seizure detection techniques.

regarding these error measures. This figure also shows that using only the onset for seizure detection is not the best possible solution. If a lower detection delay is required, it is best to lower the threshold as opposed to only use the seizure onset in the training data. If these results are compared with the results achieved using the SVM method, it is clear that the RC based method has a comparable performance. There is no clear winner between the two methods. However, note that the detection delay is not measured for missed seizures.

The detection delay of the RC-based method can be reduced by

**Table 5.4:** The average number of FPPS, FNPS and detection delay together with the standard deviation (between brackets) for the methods tested in this chapter. Most methods were personally implemented and tested on the datasets described in Section 5.1. However, the results of SVM-O* (it represents the SVM-O method as reported in Shoeb (2009)) and the results of the Reveal algorithm have been taken from Shoeb (2009), where they were tested on a slightly different dataset.

| **Scalp** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| RC | 0.43 (0.59) | 0.10 (0.15) | 14 (11) |
| RC-O | 1.5 (5.3) | 0.08 (0.14) | 9.5 (4.0) |
| RC-E | 0.52 (0.49) | 0.08 (0.12) | 11 (10) |
| RC-C | 5.9 (8.0) | 0.28 (0.41) | -35 (82) |
| RC-AL | 0.5 (1.1) | 0.36 (0.43) | -50 (160) |
| RC-AL+ | 1.2 (1.7) | 0.13 (0.19) | -190 (280) |
| SVM | 0.28 (0.40) | 0.28 (0.35) | 16 (12) |
| SVM-O | 0.11 (0.23) | 0.54 (0.42) | 13 (7) |
| SVM-O* | 0.51 (0.62) | 0.09 (0.14) | 4.6 (2.8) |
| Reveal | 18 (36) | 0.35 (0.35) | n/a |

| **iEEG** | FPPS | FNPS | $\Delta_{delay}$ |
|---|---|---|---|
| RC | 2.2 (3.6) | 0.32 (0.32) | 15 (12) |
| RC-O | 2.7 (5.3) | 0.27 (0.28) | 13 (9) |
| RC-E | 1.2 (2.1) | 0.27 (0.29) | 16 (13) |
| RC-C | 2.9 (7.0) | 0.39 (0.39) | -80 (460) |
| RC-AL | 0.65 (0.91) | 0.41 (0.39) | -170 (410) |
| RC-AL+ | 7 (13) | 0.18 (0.27) | -440 (660) |
| SVM | 0.8 (2.8) | 0.48 (0.41) | 24 (23) |
| SVM-O | 0.03 (0.11) | 0.69 (0.36) | 15 (10) |
| OFA | 32 (49) | 0.24 (0.34) | 21 (14) |

lowering the high threshold. If for example 4 FPPS are allowed, 98% of the seizures get detected with an average detection delay below 6 seconds for the Scalp dataset. For the iEEG dataset 76% of the seizures appear to be predicted by on average 60 seconds. However, seizure prediction is a very controversial patient (Mormann et al., 2007) and as shown in Figure 1.10 and 1.11, seizures in the iEEG dataset are not always marked in exactly the same way. Therefore this type of prediction is usually referred to as early seizure detection.

The results achieved using RC are very similar to the results achieved in Shoeb (2009) as shown in Table 5.4. The average number of FPPS is 0.43 and 0.51 for the RC based and SVM based methods respectively and the percentage of missed seizures are 9.7 and 8.9 %. From this point of view both methods are equivalent with respect to the performance. However, the detection delay for the RC based method is on average 14 seconds with the time to compute the preprocessing windows included, whereas in Shoeb (2009) the SVM based method has been reported to achieve an average latency of 4.6 seconds. As mentioned earlier, it remains unclear as to why the implementation of the SVM method used in this work is unable to achieve similar performance.

## 5.6 Early seizure detection

The previous section shows that some seizures can be detected before the onset is marked on the EEG. This can indicate that the seizure started some time before the onset was marked. Since the beginning and the end of the seizures can be wrongly marked it might be best to ignore the desired output right before and after the markings. To test this assumption the training set is altered. If a one hour training example contains a seizure, the non-seizure output 10 minutes before and after the seizure is ignored. For convenience and to avoid overfitting on input features related to electrical artefacts, no other output data is ignored. Since a reservoir is a dynamical system with a fading memory, the input data was not omitted. The results of this experiment are shown in Figures 5.4 and 5.5 and in Table 5.4 (marked with

E). In these results, detections within a 2 minutes pre-seizure window are labelled as true positives. Since detections on the inter-ictal period after a seizure would indicate that another seizure is imminent, these are marked as false positives.

For the Scalp dataset one can conclude that this approach is able to detect 92% of the seizures with an average detection delay of 11 seconds, while detecting about 1 false positive for every 3 detections. This is a near status-quo for the number FPPS and FNPS, only the detection delay is slightly reduced. From all the patients there are 3 patients for whom some of the seizures are detected before the marked seizure onset. For patient 8, only 1 out of 5 seizures is detected before the marked onset, for patient 10, 1 in 7 and for patient 11, 2 of the 3 seizures.

In about one third of the patients there are, on average, 10 times more false positives detected before the seizure onset. Some of those false positives are related to artefacts, some are not visually distinguishable, but most seem to detect some form of epileptic non-seizure activity. This indicates that for those patients a seizure is preceded by inter-ictal bursts. If all detections before a seizure, in a one hour file containing a seizure, are considered as true positives, there are 15 patients for whom some of the seizures are detected before the seizure onset. The first detection within this window is often a signal artefact or a previously detected seizure. If previously detected seizures are ignored, at least one seizure is detected before the seizure onset in 12 patients. Since the artefacts are not consistently marked on the data, it is hard to argue when detections are related to false positives and when they are truly early detected seizures. However, only in patient 11 and 13 more than 1 seizure was detected before the marked seizure onset and only for patient 11 all seizures were detected, on average, 13 minutes before the marked seizure onset. For this patient no false positives or missed seizures were recorded.

For the iEEG dataset, 73% of the seizures get detected with an average detection delay of 16 seconds and 1.2 FPPS, as opposed to 68% with an average detection delay of 15 seconds and 2.2 FPPS for the normal patient specific model. The increase in the detection delay over the previous section, is mainly due to patient 13, for whom the system is now able to detect the seizures. On the iEEG dataset there

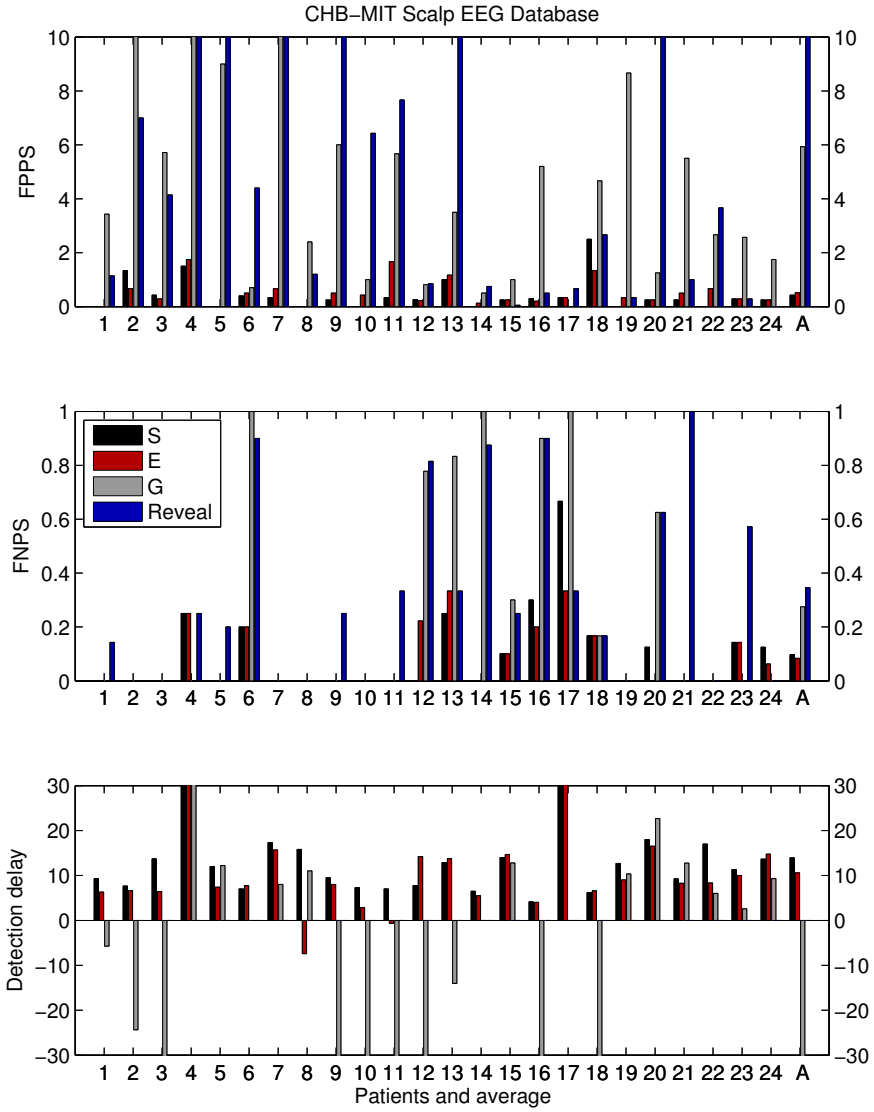


**Figure 5.4:** The FPPS, FNPS and detection delay for the RC-based patient specific seizure detection (S), early seizure detection (E) and general seizure detection technique (G) together with the Reveal algorithm on the Scalp dataset. The number of FPPS is cut off at 10 and the detection delay at -30 and 30 seconds.

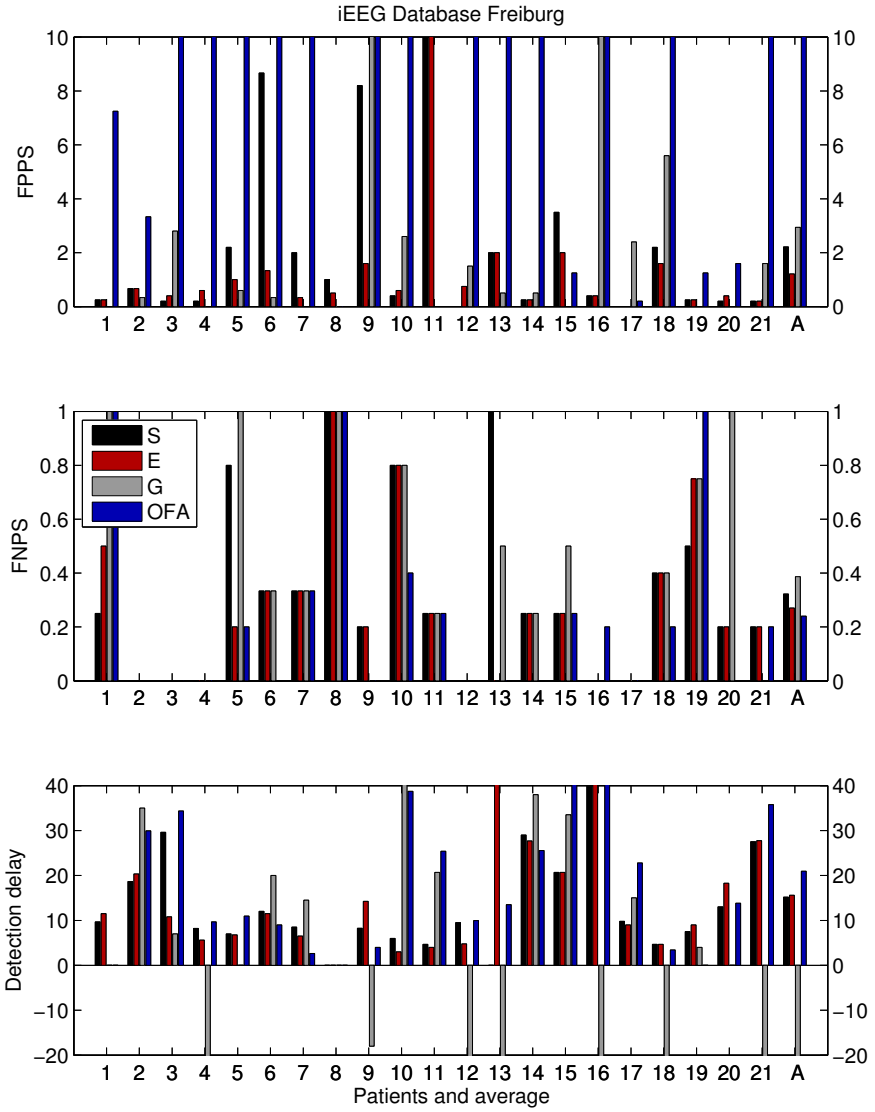


**Figure 5.5:** The FPPS, FNPS and detection delay for the RC-based patient specific seizure detection (S), early seizure detection (E) and general seizure detection technique (G) together with the Osorio-Frei algorithm (OFA) on the iEEG dataset. The number of FPPS is cut off at 10 and the detection delay at -20 and 40 seconds.

were no patients for whom seizures were detected before the seizure onset.

In one third of the patients there are 3.5 times more false positives before the seizure onset. If all detections before a seizure, in a one hour file containing a seizure, are considered as true positives, there are 4 patients for whom seizures get detected before the marked seizure onset. Only for patient 10 more than one seizure is detected before the marked seizure onset: 2 of the 3 seizures are detected, on average, 18 minutes before the seizure onset.

## 5.7 General seizure detector

In the previous chapter a seizure detector was designed that worked on more than one animal. This section analyses whether a similar set-up can be attained on human data. To test this, the system is trained on all the data, from all the patients except for one. This is repeated for each of the patients. To train the readout weights, the covariance matrices are divided by the number of hours in each dataset, such that each patient has the same influence. The results of this experiment are shown in Figures 5.4 and 5.5 and in Table 5.4.

For the Scalp dataset it shows that 5.9 FPPS are detected as opposed to 0.5 for the early seizure detection model. The number of missed seizures also dramatically increases from about 10% to 28%. Using this model, the seizures are detected, on average, 35 seconds before the marked seizure onset. Some seizures are detected early in 12 of the 24 patients which is 10 patients more than for the early seizure detection model. For 9 patients, more than 1 seizure was detected before the marked seizure onset, but for none of the patients, all seizure were detected before the marked seizure onset. Since the number of FPPS is more than 10 times higher than for the early seizure detection model, it is hard to argue whether it consists of true early detections or false positives. Note however that only seizures, occurring within 2 minutes before the marked seizure onset, are considered as true positives. If we compare this to the performance of the Reveal algorithm (Wilson et al., 2004), it is clear that this algorithm

performs significantly better. The Reveal algorithm detects 18 FPPS and misses about 35% of the seizures.

If the performance on the individual patients is compared, it shows that for 3 of the 24 patients no seizures were detected. An example of such a patient is patient 6. As shown in Figure 1.7 this patient has partial seizures with epileptiform discharges that are unlike those of other patients. In fact, inter-ictal EEG of this patient more closely resembles the seizure activity of other patients. In 17 of the patients, the number of missed seizures is less than 50%. For these patients only 3% of the seizures are missed and the average detection delay is -20 seconds. However, the average number of FPPS seizure is 7.6 for these patients. If patients 2, 4 and 7, who have more than 10 FPPS, are also ignored the number of FPPS is reduced to a more reasonable 4, without changing the average number of missed seizures and the detection delay. This means that the RC-based general seizure detector performs somewhat reasonable for 14 out of the 24 patients on this specific dataset. Using the same selection criteria for the Reveal algorithm, 12 patients are removed. For the other patients the algorithm detects 3 FPPS and misses 11% of the seizures. This indicates that the performance of both methods is comparable for those patients with a reasonable performance, but there are more patients for whom reasonable performance is achieved for the RC-based model.

On the iEEG dataset, the general seizure detector misses 39% of the seizures, which is very similar to the performance of the patient specific model but worse than the early seizure detection model. The number of FPPS is 3. This is only twice as high as the early seizure detection model and again comparable to that of the patient specific model. The average detection delay is reduced to -80 seconds as opposed to 16 seconds. This low delay is mainly caused by some seizures, in 7 of the 21 patients, that are detected before the marked seizure onset. In 3 of these patients more than one seizure was detected before the marked seizure onset, but for none of the patients all seizures were detected before the marked seizure onset. The Osorio-Frei algorithm (OFA) (Osorio et al., 1998) detects 76% of the seizures with an average detection delay of 21 seconds. To achieve this performance 32 FPPS were detected. This is significantly worse than the RC-based

method.

If the same strategy as above is used to select individual patients from the dataset, the RC-based general seizure detector has a somewhat acceptable performance in 13 of the 21 patients. For these patients, 80% of the seizures were detected with an average detection delay of -2 minutes while detecting 1.2 FPPS. Using the same analogy, the OFA algorithm has an acceptable performance in only 5 of the 21 patients. On these patients it is able to detect 90% of the seizures with an average detection delay of 27 seconds while detecting 1.3 FPPS.

Although the RC-based general model and the methods from literature are patient unspecific, a side note needs to be made. The RC-based method is trained on the EEG data from patients where the EEG is measured on exactly the same locations. The methods from literature on the other hand work independently from the channels selected. This might explain why the RC-based method performs significantly better than the methods from literature and should be investigated further.

Although SVMs have been shown to achieve similar performance to RC in section 5.5, they have several disadvantages. To build a general model for seizure detection would require the full dataset to fit in the memory of a computer. Since it contains around $3 \cdot 10^6$ data points and more than 400 features this is not possible with current computer hardware. There are techniques that do not require the full dataset to fit in memory (Bottou, 2007), but they still require the Gram matrix to be stored in memory. The number of elements in this matrix is the square of the number of support vectors. This can again be near the limit of current computer hardware and training and executing the model becomes a significant computational burden. RC on the other hand can be trained in a much shorter time frame using the algorithms of Chapter 3 and only requires around $N^2$ weights to be stored in system memory, where $N = 1000$ represents the number of neurons in the reservoir used in this work. The number of neurons in a reservoir is usually significantly smaller than the number of support vectors for such large datasets. This means that the computational cost to execute a reservoir based model is significantly lower than that of an SVM.

## 5.8 Active learning

As mentioned in Section 4.9, active learning (AL) is usually implemented in a way that during training an expert's opinion is asked in case of doubt. For seizure detection this would mean that when the system is applied in a real-life situation, an encephalographer needs to be on standby. However, if AL is used to build a patient specific training set, for already recorded data, this approach will significantly reduce the costly expert time required to achieve similar performance (Balakrishnan and Syed, 2012).

A better approach still is not to require an expert at all. In the previous section it was shown that a seizure detector can be trained on data from other patients. This approach however does not yield the same performance as the patient specific approach and is especially characterized by a high number of FPPS. This unwanted side effect has only one useful attribute: epilepsy patients (and caregivers) are usually very well aware of the fact that they are not having a seizure. By providing the user with a device to cancel a seizure alarm, an EEG recording can be marked. This marking can then be used to add an extra example to the training set. To better fit the patient's data, true positives can also be added to the training data. This form of AL is in fact a form of supervised transfer learning.

To test this form of AL the general model for seizure detection is used as a base model. To train the reservoir output, all training data from other patients is used together with the data marked using AL. The threshold is trained using only patient specific training examples if it contains both positives and negatives. If it does not, the threshold of the general model is used instead. The experimental results are shown in Figures 5.6 and 5.7, and in Table 5.4 (marked with AL). These show that AL significantly reduces the number of false positives to a very reasonable 0.5 FPPS for the Scalp dataset and 0.65 FPPS for the iEEG dataset. The detection delay slightly decreases and the number of missed seizures is increased from 28% to 36% for the Scalp dataset and from 39% to 41% for the iEEG dataset.

If only patients for whom at least than 50% of the seizures are detected, are considered, the system has an acceptable performance

in 16 of the 24 patients on the Scalp dataset. For these patients 8% of the seizures are missed while detecting less than 0.6 FPPS. In 4 of these patients some seizures get detected before the seizure onset. For the iEEG dataset 14 of the 21 patients have an acceptable performance where, on average, 16% of the seizures are missed and 0.7 FPPS get detected. In 5 of these patients some seizures get detected before the seizure onset.

This shows that AL results in a system with a performance that is comparable to a patient specific seizure detector in about 70% of the patients. Patients for whom the general model is able to detect at least 50% of the seizures, independent of the number of FPPS, tend to result in an acceptable performance.

Because of the high number of missed seizures, it might be opportune to add missed seizures to the training set. Finding the seizures on the EEG can be done by an expert. However, caregivers are often aware when patients are having a seizure and in many cases caregivers and patients are aware that a seizure has just occurred. At these points in time they can for example push a button to indicate there is or has been a seizure. Using this information one can go back in time through the EEG, if necessary, and include the samples that are above the low threshold as ictal EEG to the training set. The results of this experiment is shown in Figures 5.6 and 5.7 and in Table 5.4, marked with AL+. This significantly reduces the number of FNPS for both datasets, and even below that of the patient specific model for the iEEG dataset. It also increases the number of FPPS over the previously discussed form of AL: to 1.2 for the Scalp dataset and to 7.9 for the iEEG dataset. This high number for the iEEG dataset is mainly caused by patient 8, for whom none of the seizure detection methods are able to detect any seizures. Without this patient the average number of FPPS on the iEEG dataset is reduced to a very reasonable 2.4. The average detection delay is even further reduced.

If only patients for whom less than 10 FPPS or at least 50% of the seizures are detected, are considered as patients with an acceptable performance, this leaves 23 of the 24 patients on the Scalp dataset. For these patients only 10% of the seizures are missed and 1 FPPS is detected. In 13 of these patients seizures get detected before the

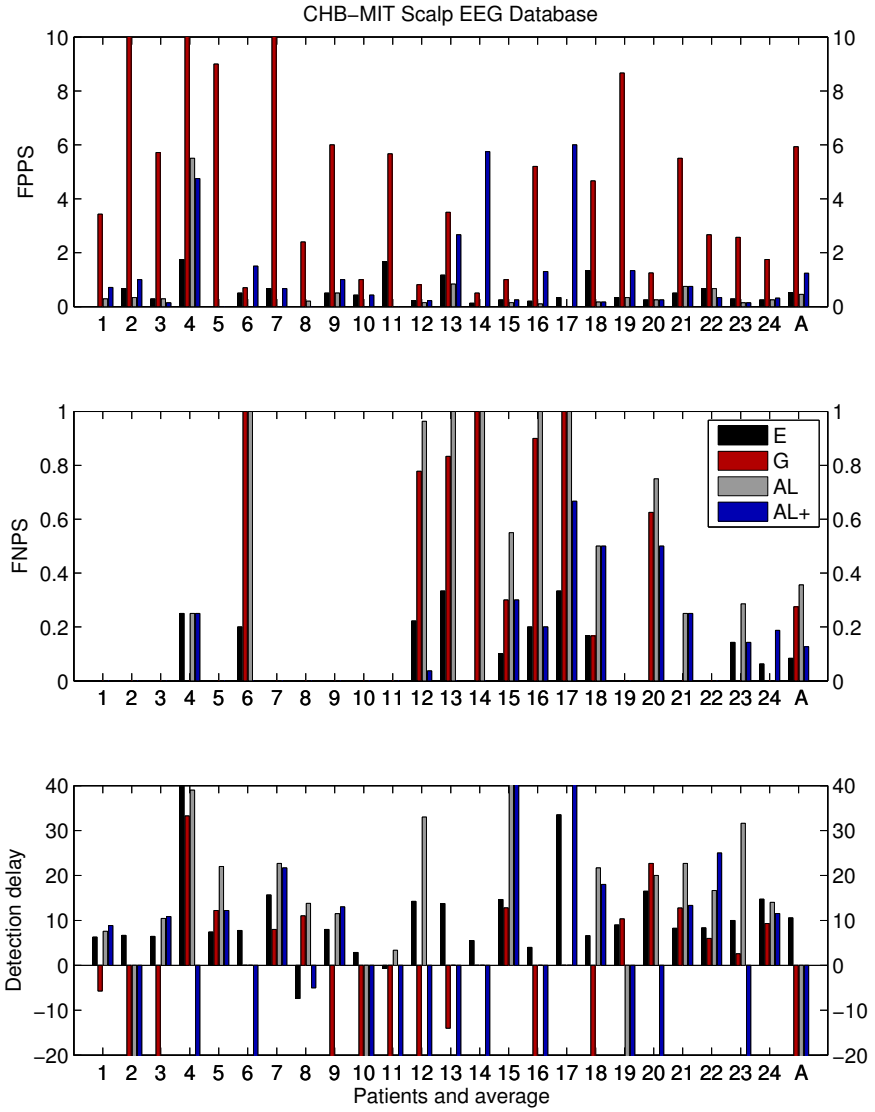


**Figure 5.6:** The FPPS, FNPS and detection delay for the RC-based early seizure detection (E) and general seizure detection technique (G) compared with the AL experiments on the Scalp dataset. The number of FPPS is cut off at 10 and the detection delay at -20 and 40 seconds.

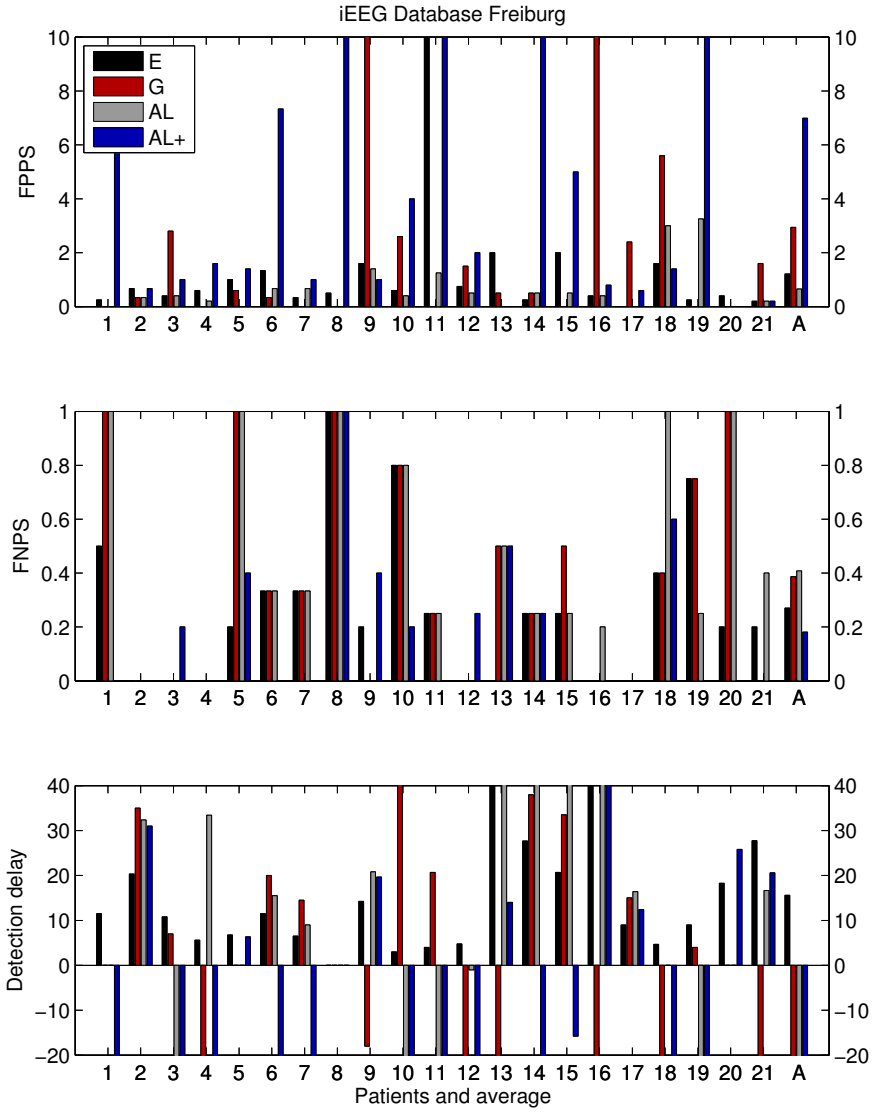


**Figure 5.7:** The FPPS, FNPS and detection delay for the RC-based early seizure detection (E) and general seizure detection technique (G) compared with the AL experiments on the iEEG dataset. The number of FPPS is cut off at 10 and the detection delay at -20 and 40 seconds.

seizure onset. This shows that this type of AL achieves a very acceptable performance on the Scalp dataset. For the iEEG dataset, acceptable performance is only attained in 13 of the 21 patients. For these patients only 8% of the seizures are missed and 1.3 FPPS get detected. In 12 of these patients some seizures are detected before the seizure onset.

Although this type of AL causes many FPPS, it can be considered an accessible technique for building a patient specific seizure detector since, as shown earlier, the high number of FPPS is easily corrected using continuous AL. If the number of FPPS are ignored it gives a comparable performance to the patient specific model in more than 90% of the patients. This number might even increase if the system were to be used over longer periods of time since the performance of the patient specific model is achieved if all data is used and not just the data marked by AL.

## 5.9 Conclusions

In this chapter it was shown that the RC-based seizure detector has a comparable performance to the SVM-based seizure detector presented in Shoeb (2009) which is currently considered the state-of-the-art in seizure detection. Both these methods were validated on 1721 hours of scalp or intra-cranial EEG from 45 patients in total, containing 272 seizures.

Based on the results of this experiment, a patient specific early seizure detection algorithm was proposed that can detect some of the seizures before the marked seizure onset in 2 of the 45 patients. On average, a detection delay of 13 seconds was achieved while missing only 16% of the seizures and detecting less than 1 false positive for every 2 detections. This performance is a significant reduction in the number of FPPS over the normal RC seizure detector, as well as a slight reduction in the number of missed seizures and the detection delay.

A non patient specific seizure detector was proposed that was able to detect some seizures before the marked seizure onset in 17 of the 45

tested patients. It missed 33% of the seizures and detected on average 4.5 FPPS. This is a clear improvement over seizure detectors from literature (Osorio et al., 1998; Wilson et al., 2004), but it is significantly outperformed by the patient specific early seizure detection model. For this task RC has a clear advantage over SVMs: it has a lower computation time and requires significantly less system memory, such that it can be trained and used on a regular computer.

To build a patient specific seizure detector without the need for large amounts of marked data, 2 learning strategies were proposed. These strategies build on the non patient specific seizure detector and were tested retrospectively using virtual users. The first strategy requires the user to respond when a false positive has been detected. Using this approach, the performance of the patient specific seizure detector was approximated in 70% of the patients. These patients correspond roughly to the patients for whom the non patient specific model was able to detect more than 50% of the seizures. The second strategy required the user to also press a button during or right after a seizure. Using this approach, the performance of the patient specific model was achieved in more than 90% of the patients, albeit with a higher number of false positives. This higher number of false positives can be easily overcome by applying one of both approaches continuously.

# 6 Conclusions and future prospects

## 6.1 Summary and conclusions

Because in epileptic seizure detection it is hard to define one error measure that needs to be optimized, and because usually very large datasets are used, several new algorithms have been proposed in this work to train recurrent neural networks using the reservoir computing approach. These algorithms have been designed in such a way that their computational cost and memory requirements are limited. The performance of these algorithms has been evaluated and compared to the state-of-the-art for epileptic seizure detection in animal models. In these experiments the Bayesian relevance regression algorithm gave the best performance. This algorithm automatically weighs the influence of each seizure example according to its relevance to train a general seizure detection model.

For animal models, a non rat specific seizure detection algorithm has been proposed that achieves state-of-the-art performance. It has been validated on data sets of two different animal models: the genetic altered epilepsy rats from Stassbourg (GAERS) and the post status epilepticus (PSE) model. The performance achieved has been shown to be comparable to that of some human encephalographers. This allows for the system to be used as a tool to automatically mark epileptic seizures on the EEG and as an on-line seizure detector for research towards closed loop anti-epileptic treatment.

Every seizure detector is confronted with a detection delay. For the RC-based method, this delay and the number of missed seizures can be traded of against the number of false positives by altering a threshold parameter. This allows for a lower response time and fewer missed seizures in closed loop experiments at the cost of more FPPS. This approach can also be applied for highly accurate seizure marking and will reduce the workload of the encephalographer by up to 90%. To even further reduce the workload, several learning strategies have been proposed that reduce the amount of required training data.

For human epilepsy patients a patient specific early seizure detection system has been proposed that has a performance which is comparable to the current state-of-the-art. It has been validated on scalp and intra-cranial EEG from 45 patients in total. A non patient specific seizure detector has been proposed that outperforms the methods used in literature, but which is unable to reach the performance of the patient specific model. In some of the patients, these models were able to detect seizures before the marked seizure onset.

To build a patient specific seizure detector without the need for large amounts of marked data, two active learning strategies have been proposed. They allow the performance of the non patient specific seizure detector to be adapted to the patient without requiring experienced encephalographers and can be implemented by simple button presses. If the patient and/or caregiver only indicate when a false positive is detected, the performance of the patient specific model is attained in 70% of the patients. This can be increased to over 90% when the user is able to indicate when a seizure was missed during or right after a seizure.

## 6.2 Future prospects

As with any research, this work is not finished. Based on the results achieved in this work, there are many directions that form interesting research topics. Those that I consider most relevant are given below.

- Since state-of-the-art performance has been achieved on both the animal and the human data, the system proposed in this

work might be suited for anti-epileptic treatment in an on-line setting. And possibly it has the ability to stimulate the research on such treatment, which is still at a very early stage of development.

- For the animal models, long term prospective evaluation has been performed. The system was trained on the first hours of EEG of the dataset, and was tested on weeks of data following this training set. As the human datasets used span a much shorter period, no such experiments have been done using the human seizure detection model. This is however important to show the performance of the algorithms in a real life setting.

- In this work it has been shown that it is possible to detect some seizures before the marked seizure onset in some of the human patients. These results were achieved by training the system for seizure detection as opposed to seizure prediction. This shows that the pre-seizure state might be similar to the seizure state, and can indicate that there is electrical seizure activity present in the brain long before the seizure onset. Where this activity comes from and how it is generated remains unknown and further research could shed a new light on the underlying process of epilepsy.

- Although it has been shown that it is possible to detect seizures before the marked seizure onset for some patients in the human dataset, no such experiments have been conducted on the animal dataset yet. For epilepsy research, and for research in animal models in general, it would be of great value if similar results could be achieved in animal models. Although such work exists in literature (Nandan et al., 2010), these algorithms typically have up to 100 FPPS.

- Some research (Mirowski et al., 2009) shows, albeit inconclusively, that it is possible to predict seizures about 60 minutes before the seizure onset. It would be interesting to reproduce these results using reservoir computing, without selecting the best model on the test set as was done in Mirowski et al. (2009). The features used in this work have been shown to be suited

for (early) seizure detection. Possibly, other features that have been shown to contain predictive capabilities (Mormann et al., 2007; Mirowski et al., 2009), could allow the system to predict epileptic seizures.

- Although EEG is the most frequently used signal for detecting epileptic seizures, other signals have been used in literature such as electrocardiogram (ECG) (Zijlmans et al., 2002), accelerometry (Nijsen et al., 2005) and even skin conductance (Poh et al., 2012). A combination of these features with EEG can improve the seizure detection accuracy (Shoeb, 2009; Conradsen et al., 2009) and could be tested in the RC-based set-up.

- Measuring the EEG in a home environment is rather impractical. The most commonly used EEG devices require to constantly wear a cap which contains a conductive gel. This is not only considered uncomfortable but also unfashionable. Although implantable EEG devices with subcutaneous[1] or intra-cranial[2] electrodes have been developed, they require a surgical procedure that might hold significant risks for the patient if intra-cranial electrodes are used. On the other hand if accelerometers, gyroscopes, ECG and/or EMG electrodes could be used for seizure detection, in combination with the user-based learning strategies proposed in this work, a much more compelling seizure detector could be built for the patient.

[1] http://www.hyposafe.com/
[2] http://www.neuropace.com/

# Bibliography


Allen, D. (1974). The Relationship Between Variable Selection and Data Augmentation and Slow Feature Analysis. *Technometrics*, 16.

Angeles, D. (1981). Proposal for revised clinical and electroencephalographic classification of epileptic seizures. from the commission on classification and terminology of the international league against epilepsy. *Epilepsia*, 22:489–501.

Balakrishnan, G. and Syed, Z. (2012). Scalable personalization of long-term physiological monitoring: Active learning methodologies for epileptic seizure onset detection. *Journal of Machine Learning Research*, 22:73–81.

Baraban, S., editor (2009). *Animal Models of Epilepsy: Methods and Innovations*, volume 40 of *Neuromethods*. Humana Press, New York.

Benbadis, S. and Allen Hauser, W. (2000). An estimate of the prevalence of psychogenic non-epileptic seizures. *Seizure*, 9(4):280–281.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer.

Boon, P., Vonck, K., De Reuckb, J., and Cammaert, J. (2001). Vagus nerve stimulation for refractory epilepsy. *Seizure*, 10:448–455.

Bottou, L. (2007). *Large-scale kernel machines*. Mit Pr.

Buteneers, P., Caluwaerts, K., Verstraeten, D., and Schrauwen, B. (2012a). Optimized parameter search for large datasets of the regularization parameter and feature selection for ridge regression. *Neural Processing Letters*. (under revision).

Buteneers, P., Verstraeten, D., Nieuwenhuyse, B., Stroobandt, D., Raedt, R., Vonck, K., Boon, P., and Schrauwen, B. (2012b). Real-time detection of epileptic seizures in animal models using reservoir computing. *Epilepsy Research*. (in press).

Buteneers, P., Verstraeten, D., van Mierlo, P., Wyckhuys, T., Staelens, S., Stroobandt, D., and Schrauwen, B. (2010). Automatic Detection of Epileptic Seizures on Intra-cranial EEG from Rats using Reservoir Computing. *AI in Medicine*, 53:215–223.

Cawley, G. and Talbot, N. (2004). Fast exact leave-one-out cross-validation of sparse least-squares support vector machines. *Neural Networks*, 17:1467–1475.

Chauvin, Y. and Rumelhart, D. (1995). *Backpropagation: theory, architectures, and applications*. Lawrence Erlbaum.

Conradsen, I., Beniczky, S., Wolf, P., Terney, D., Sams, T., and Sorensen, H. (2009). Multi-modal intelligent seizure acquisition (misa) systemÑa new approach towards seizure detection based on full body motion measures. In *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pages 2591–2595. IEEE.

Coone, A. (2011). A study on different preprocessing and machine learning techniques for the detection of error-potentials in brain-computer interfaces. Master's thesis, Ghent University.

Costa, R., Oliveira, P., Rodrigues, G., Leitão, B., and Dourado, A. (2008). Epileptic seizure classification using neural networks with 14 features. In Lovrek, I., Howlett, R., and Jain, L., editors, *Knowledge-Based Intelligent Information and Engineering Systems. 12th International Conference, KES 2008*, pages 281–288, Zagreb, Croatia. University of Zagreb, Springer.

Cover, T. (1965). Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. *IEEE transactions on electronic computers*, 14(3):326–334.

Cysyk, B., Sepkuty, J., Lesser, R., and Civelek, A. (1997). Truly ictal spect is of major importance for reliable localization of seizure focus. *Epilepsia*, 38(Supplement 8):146.

de Boer, H., Mula, M., and Sander, J. (2008). The global burden and stigma of epilepsy. *Epilepsy & behavior*, 12(4):540–546.

De Brabanter, K., De Brabanter, J., Suykens, J., and De Moor, B. (2010). Optimized fixed-size kernel models for large data sets. *Computational Statistics & Data Analysis*, 54:1484–1504.

Dedeurwaerdere, S. (2005). *Neuromodulation in experimental animal models of epilepsy*. PhD thesis, Ghent University.

Drachman, D. (2005). Do we have brain to spare? *Neurology*, 64(12):2004–2005.

Duncan, J., Sander, J., Sisodiya, S., Walker, M., et al. (2006). Adult epilepsy. *Lancet (London, England)*, 367(9516):1087–1100.

Dutoit, X., Schrauwen, B., Van Campenhout, J., Stroobandt, D., Van Brussel, H., and Nuttin, M. (2009). Pruning and regularization in reservoir computing. *Neurocomputing*, 72:1534–1546.

Efron, B., Hastie, T., Johnstone, I., and Tibshirani, R. (2004). Least angle regression. *The Annals of Statistics*, 32:407–451.

Fanselow, E., Ashlan, P., and Nicolelis, A. (2000). Reduction of pentylenetetrazole-induced seizure activity in awake rats by seizure-triggered trigeminal nerve stimulation. *Journal of Neuroscience*, 20:8160–8168.

Fernando, C. and Sojakka, S. (2003). Pattern recognition in a bucket. In *Proceedings of the 7th European Conference on Artificial Life*, pages 588–597.

Gallagher Jr, N. and Wise, G. (1981). A theoretical analysis of the properties of median filters. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29(6):1136–1141.

Gardner, A., Krieger, A., Vachtsevanos, G., and Litt, B. (2006). One-class novelty detection for seizure analysis from intracranial eeg. *The Journal of Machine Learning Research*, 7:1025–1044.

Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P., Mark, R., Mietus, J., Moody, G., Peng, C., and Stanley, H. (2000). Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220.

Golub, G. and Van Loan, C. (1989). *Matrix computations*. The Jonhs Hopkins University Press.

Gotman, J. (1982). Automatic recognition of epileptic seizures in the eeg. *Electroencephalography and clinical Neurophysiology*, 54(5):530–540.

Gotman, J., Ives, J., and Gloor, P. (1981). Frequency content of eeg and emg at seizure onset: possibility of removal of emg artefact by digital filtering. *Electroencephalography and clinical neurophysiology*, 52(6):626–639.

Grewal, S. and Gotman, J. (2005). An automatic warning system for epileptic seizures recorded on intracerebral eegs. *Clinical neurophysiology*, 116(10):2460–2472.

Guyon, I. and Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3:1157–1182.

Haas, S., Frei, M., and Oso (2007). Strategies for adapting automated seizure detection algorithms. *Medical Engineering & Physics*, 29:895–909.

Hammond, E., Uthman, B., Reid, S., and Wilder, B. (1992). Electrophysiological studies of cervical vagus nerve stimulation in humans: I. eeg effects. *Epilepsia*, 33(6):1013–1020.

Herculano-Houzel, S. (2009). The human brain in numbers: a linearly scaled-up primate brain. *Frontiers in human neuroscience*, 3.

Hermans, M. and Schrauwen, B. (2010). Memory in linear recurrent neural networks in continuous time. *Neural Networks*, 23(3):341–355.

Hermans, M. and Schrauwen, B. (2012). Recurrent kernel machines: Computing with infinite echo state networks. *Neural Computation*, 24(1):104–133.

Holland, P. (1973). Weighted Ridge Regression: Combining Ridge and Robust Regression Methods. *Technical Report 0011, National Bureau of Economic Research, Inc.*

Homan, R., Herman, J., and Purdy, P. (1987). Cerebral location of international 10-20 system electrode placement. *Electroencephalography and Clinical Neurophysiology*, 66(4):376–382.

Hoppe, C., Poepel, A., and Elger, C. (2007). Epilepsy: accuracy of patient seizure counts. *Archives of neurology*, 64(11):1595.

Hsu, C., Chang, C., Lin, C., et al. (2003). A practical guide to support vector classification.

ILAE (1989). Proposal for revised classification of epilepsies and epileptic syndromes. *Epilepsia*, 30:389–399. Commission on Classification and Terminology of the International League Against Epilepsy.

Jaeger, H. (2001). Short term memory in echo state networks. Technical Report GMD Report 152, German National Research Center for Information Technology.

Jaeger, H. (2002). Tutorial on training recurrent neural networks, covering BPTT, RTRL, EKF and the "echo state network" approach. Technical Report GMD Report 159, German National Research Center for Information Technology.

Jaeger, H., Lukosevicius, M., and Popovici, D. (2007). Optimization and applications of echo state networks with leaky integrator neurons. *Neural Networks*, 20:335–352.

Jirsch, J., Urrestarazu, E., LeVan, P., Olivier, A., Dubeau, F., and Gotman, J. (2006). High-frequency oscillations during human focal seizures. *Brain*, 129(6):1593–1608.

Kohavi, R. et al. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *International joint Conference on artificial intelligence*, volume 14, pages 1137–1145. Lawrence Erlbaum Associates Ltd.

Krogh, A. and Vedelsby, J. (1995). Neural network ensembles, cross validation, and active learning. *Advances in neural information processing systems*, pages 231–238.

Legenstein, R. A. and Maass, W. (2007). Edge of chaos and prediction of computational performance for neural microcircuit models. *Neural Networks*, pages 323–333.

Levine, H. (1997). Rest heart rate and life expectancy. *Journal of the American College of Cardiology*, 30(4):1104.

Lowenstein, D. and Alldredge, B. (1998). Status epilepticus. *New England Journal of Medicine*, 338(14):970–976.

Lukoševičius, M. and Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. *Computer Science Review*, 3(3):127–149.

Marescaux, C., M., V., and A., D. (1992). Genetic Absence Epilepsy in Rats from Strasbourg - A Review. *Journal of Neural Transmission*, 35:37–69.

Menard, S. (2002). *Applied logistic regression analysis*, volume 106. Sage Publications, Inc.

Miche, Y., Sorjamaa, A., Bas, P., Simula, O., Jutten, C., and Lendasse, A. (2010). Op-elm: Optimally pruned extreme learning machine. *IEEE Transactions on Neural Networks*, 21:158–162.

Mirowski, P., Madhavan, D., LeCun, Y., and Kuzniecky, R. (2009). Classification of patterns of eeg synchronization for seizure prediction. *Clinical neurophysiology*, 120(11):1927–1940.

Moore, F. (1989). The desperate case: Care (costs, applicability, research, ethics). *JAMA: the journal of the American Medical Association*, 261(10):1483–1484.

Mormann, F., Andrzejak, R., Elger, C., and Lehnertz, K. (2007). Seizure prediction: the long and winding road. *Brain*, 130(2):314–333.

Nandan, M., Talathi, S., Myers, S., Ditto, W., Khargonekar, P., and Carney, P. (2010). Support vector machines for seizure detection in an animal model of chronic epilepsy. *Journal of Neural Engineering*, 7(3):036001.

Nijsen, T., Arends, J., Griep, P., and Cluitmans, P. (2005). The potential value of three-dimensional accelerometry for detection of motor seizures in severe epilepsy. *Epilepsy & Behavior*, 7(1):74–84.

Nijsen, T., Cluitmans, P., Arends, J., and Griep, P. (2007). Detection of subtle nocturnal motor activity from 3-d accelerometry recordings in epilepsy patients. *Biomedical Engineering, IEEE Transactions on*, 54(11):2073–2081.

Ojeda, F., Suykens, J., and De Moor, B. (2008). Low rank updated LS-SVM classifiers for fast variable selection. *Neural Networks*, 21:437–449.

Osorio, I., Frei, M., Giftakis, J., Peters, T., Ingram, J., Turnbull, M., Herzog, M., Rise, M., Schaffner, S., Wennberg, R., et al. (2002). Performance reassessment of a real-time seizure-detection algorithm on long ecog series. *Epilepsia*, 43(12):1522–1535.

Osorio, I., Frei, M., and Wilkinson, S. (1998). Real-time automated detection and quantitative analysis of seizures and short-term prediction of clinical onset. *Epilepsia*, 39(6):615–627.

Pacia, S. and Ebersole, J. (1997). Intracranial eeg substrates of scalp ictal patterns from temporal lobe foci. *Epilepsia*, 38(6):642–654.

Pahikkala, T., Airola, A., and Salakoski, T. (2010). Feature selection for regularized least-squares: new computational short-cuts and fast algorithmic implementations. In *IEEE International Workshop on Machine Learning for Signal Processing*.

Pahikkala, T., Boberg, J., and Salakoski, T. (2006). Fast n-fold cross-validation for regularized least-squares. In *Proceedings of the Ninth Scandinavian Conference on Artificial Intelligence (SCAI)*.

Päivinen, N., Lammi, S., Pitkänen, A., Nissinen, J., Penttonen, M., and Grönfors, T. (2005). Epileptic seizure detection: A nonlinear viewpoint. *Computer Methods and Programs in Biomedicine*, 79:151–159.

Pan, S. and Yang, Q. (2010). A survey on transfer learning. *Knowledge and Data Engineering, IEEE Transactions on*, 22(10):1345–1359.

Parlett, B. (1980). *The symmetric eigenvalue problem*, volume 7. SIAM.

Pfurtscheller, G. and Lopes da Silva, F. (1999). Event-related eeg/meg synchronization and desynchronization: basic principles. *Clinical neurophysiology*, 110(11):1842–1857.

Pineda, F. (1987). Generalization of back-propagation to recurrent neural networks. *Physical Review Letters*, 59(19):2229–2232.

Poh, M., Loddenkemper, T., Reinsberger, C., Swenson, N., Goyal, S., Madsen, J., and Picard, R. (2012). Autonomic changes with seizures correlate with postictal eeg suppression. *Neurology*, 78(23):1868–1876.

Press, W. (1992). Numerical recipes in c: the art of scientific computing. 2nd ed. Cmbridge; New York: Cambridge University Press.

Quesney, L., Gloor, P., et al. (1985). Localization of epileptic foci. *Electroencephalography and clinical neurophysiology. Supplement*, 37:165.

Riley, J. (2001). *Rising life expectancy: a global history*. Cambridge Univ Pr.

Saab, M. and Gotman, J. (2005). A system to detect the onset of epileptic seizures in scalp eeg. *Clinical Neurophysiology*, 116(2):427–442.

Sherman, J. and Morisson, W. J. (1950). Adjustments of an Inverse Matrix Corresponding to a Change in One Element of a Given Matrix. *The Annals of Mathematical Statistics*, 21:124–127.

Shoeb, A. (2009). *Application of machine learning to epileptic seizure onset detection and treatment*. PhD thesis.

Shoeb, A., Edwards, H., Connolly, J., Bourgeois, B., Ted Treves, S., and Guttag, J. (2004). Patient-specific seizure onset detection. *Epilepsy & Behavior*, 5(4):483–498.

Stahl, V., Fisher, A., and Bippus, R. (2000). Quantile based noise estimation for spectral subtraction and Wiener filtering. In *International Conference on Acoustics Speech and Signal Processing (ICASSP)*, Istanbul, Turkey. IEEE.

Stein, A., Eder, H., Blum, D., Drachev, A., and Fisher, R. (2000). An automated drug delivery system for focal epilepsy. *Epilepsy research*, 39(2):103–114.

Tao, J., Ray, A., Hawes-Ebersole, S., and Ebersole, J. (2005). Intracranial eeg substrates of scalp eeg interictal spikes. *Epilepsia*, 46(5):669–676.

Teplan, M. (2002). Fundamentals of eeg measurement. *Measurement science review*, 2(2):1–11.

Theodore, W. and Fisher, R. (2004). Brain stimulation for epilepsy. *The Lancet Neurology*, 3(2):111–118.

Tikhonov, A. and Arsenin, V. Y. (1977). *Solutions of Ill-Posed Problems*. Winston and Sons.

Ting, J., D'Souza, A., and Schaal, S. (2007). Automatic outlier detection: A bayesian approach. In *Conference on Robotics and Automation, 2007 IEEE International*.

Tipping, M. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244.

Toh, K. A. (2008). Deterministic Neural Classification. *Neural Computation*, 20:1565–1595.

Tzallas, A., Tsipouras, M., Tsalikakis, D., Karvounis, E., Astrakas, L., Konitsiotis, S., and Tzaphlidou, M. (2012). Automated epileptic seizure detection methods: A review study. Technical report.

Urrestarazu, E., Chander, R., Dubeau, F., and Gotman, J. (2007). Interictal high-frequency oscillations (100–500 hz) in the intracerebral eeg of epileptic patients. *Brain*, 130(9):2354–2366.

Van Hese, P., Martens, J., Boon, P., Dedeurwaerdere, S., Lemahieu, I., and Van de Walle, R. (2003). Detection of spike and wave discharges in the cortical EEG of genetic absence epilepsy rats from Strasbourg. *Physics in Medicine and Biology*, 48(12):1685–1700.

Van Hese, P., Martens, J.-P., Waterschoot, L., Boon, P., and Lemahieu, I. (2009). Automatic Detection of Spike and Wave Discharges in the EEG of Genetic Absence Epilepsy Rats from Strasbourg. *IEEE Transactions on Biomedical Engineering*, 56(3):706–717.

Vandoorne, K., Dierckx, W., Schrauwen, B., Verstraeten, D., Baets, R., Bienstman, P., and Van Campenhout, J. (2008). Toward optical signal processing using photonic reservoir computing. *Optics Express*, 16(15):11182–11192.

Verstraeten, D. (2009). *Reservoir Computing : computation with dynamical systems*. PhD thesis, Ghent University.

Verstraeten, D., Dambre, J., Dutoit, X., and Schrauwen, B. (2010). Memory versus non-linearity in reservoirs. In *Neural Networks (IJCNN), The 2010 International Joint Conference on*, pages 1–8. IEEE.

Verstraeten, D., Schrauwen, B., D'Haene, M., and Stroobandt, D. (2007). A unifying comparison of reservoir computing methods. *Neural Networks*, 20:391–403.

Verstraeten, D., Schrauwen, B., Dieleman, S., Brakel, P., Buteneers, P., and Pecevski, D. (2011). Oger: Modular learning architectures for large-scale sequential processing. (in press).

Vonck, K., Boon, P., Goossens, L., Dedeurwaerdere, S., Claeys, P., Gossiaux, F., Van Hese, P., De Smedt, T., Raedt, R., Achten, E., et al. (2003). Neurostimulation for refractory epilepsy. *Acta neurologica belgica*, 103(4):212–217.

Wang, Z. and Zhang, D. (1999). Progressive switching median filter for the removal of impulse noise from highly corrupted images. *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on*, 46(1):78–80.

Waterschoot, L., Dedeurwaerdere, S., and Wyckhuys, T. (2006). Stimulation of the anterodorsal thalamus in genetic absence epilepsy rats from Strasbourg (GAERS). In Epilepsia, editor, *7th European Congress on Epileptology*, volume 47, page 70, Helsinki, Finland. Blackwell Publishing.

Westerhuis, F., Van Schaijk, W., and Van Luijtelaar, G. (1996). Automatic detection of spike-wave discharges in the cortical EEG of rats. In *Measuring Behavior '96, International Workshop on Methods and Techniques in Behavioral Research*, Utrecht, The Netherlands. Utrecht University.

White, A., Willians, P., Ferraro, D., Clark, S., Kadam, S., and Dudek et al., F. (2006). Efficient unsupervised algorithms for the detection of seizures in continuous EEG recordings from rats after brain injury. *Journal of Neuroscience Methods*, 152:255–266.

Wilson, S., Scheuer, M., Emerson, R., and Gabor, A. (2004). Seizure detection: evaluation of the reveal algorithm. *Clinical neurophysiology*, 115(10):2280–2291.

Witte, H., Iasemidis, L., and Litt, B. (2003). Special issue on epileptic seizure prediction. *Biomedical Engineering, IEEE Transactions on*, 50(5):537–539.

Wyckhuys, T., Boon, P., Raedt, R., Van Niewenhuyse, B., Vonck, K., and Wadman, W. (2010). Suppression of hippocampal epileptic seizures in the kainate rat by poisson distributed stimulation. *Epilepsia*, 51:2297–2304.

Zijlmans, M., Flanagan, D., and Gotman, J. (2002). Heart rate changes and ecg abnormalities during epileptic seizures: prevalence and definition of an objective clinical sign. *Epilepsia*, 43(8):847–854.